PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 12/00	A1	(11) International Publication Number: **WO 98/59460** (43) International Publication Date: 30 December 1998 (30.12.98)

(21) International Application Number: PCT/US98/12947

(22) International Filing Date: 22 June 1998 (22.06.98)

(30) Priority Data:
08/879,859 20 June 1997 (20.06.97) US

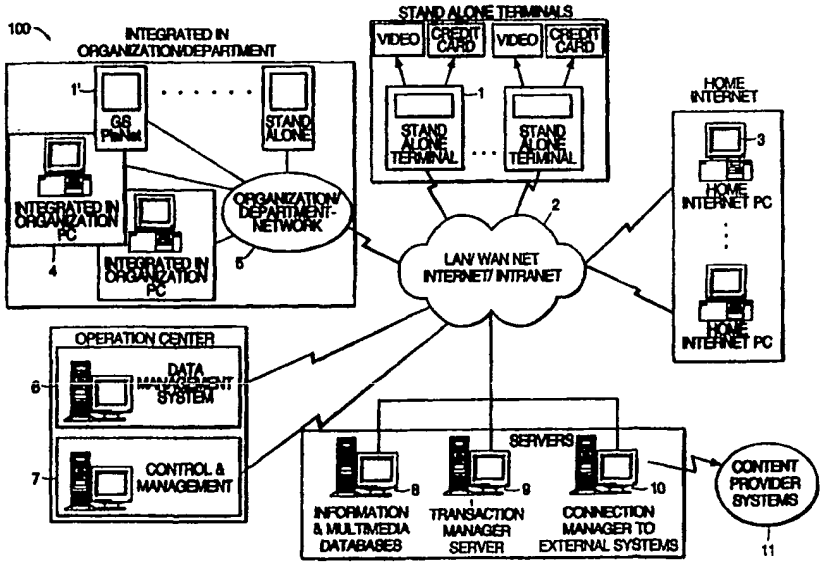
(71)(72) Applicant and Inventor: RAZ, Uri [US/US]; 36-02 Hillside Terrace, Fairlawn, NJ 07410 (US).

(74) Agent: SIBER, Victor; Fish & Richardson P.C., 45 Rockefeller Plaza, New York, NY 10111 (US).

(81) Designated States: CA, IL, JP, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: INFORMATION TRANSFER SYSTEM WITH DYNAMIC DISTRIBUTION OF DATA, CONTROL AND MANAGEMENT OF INFORMATION



(57) Abstract

A method and system for dynamically distributing information across a network (100) is provided.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

INFORMATION TRANSFER SYSTEM WITH DYNAMIC DISTRIBUTION OF DATA, CONTROL AND MANAGEMENT OF INFORMATION

Background of the Invention

The present invention relates generally to information transfer systems, and more particularly to the dynamic distribution and transfer of information over a data communications network, for example, the Internet.

The transfer of information using the Internet is not centrally managed. The Internet is an amalgam of many networks and many host computers. Using the Internet, information is stored on host servers and is requested using a static address location of that host server. The Internet does not redistribute data according to user requests. Furthermore, the Internet is organized into a hierarchy of two tiers, a client tier and a server tier. Thus, the Internet does not efficiently manage the location of information. Furthermore, the Internet does not enjoy the advantages of a four tier system, and its system resources are not managed.

The Internet is not a consistent behaving network. The traffic and load on parts of its infrastructure can largely vary. Thus, the response time is inconsistent, unstable, and unexpected. Therefore, traditional Internet tools (i.e. servers, clients) cannot be used "as is", if a professional level of interactivity is required.

Traditional Local Area Networks ("LANs") or Wide Area Networks ("WANs") that employ dedicated infrastructure gain stability and dependability based on the assumption that the users, the amount of users, and the content that ran over the network are expected. In the case of the Internet, these assumptions can not be

made, thus eliminating the use of traditional LANs or WANs.

Summary of the Invention

In one embodiment, the invention is directed to a system for transferring data over a network having a plurality of client terminals, and a plurality of servers connected to a network. An operations management system connected to the network monitors and stores characteristic data about the client terminals and the servers. An interfacing system connected to the network dynamically distributes data between the servers and the terminals as a function of the characteristic data. As a result, a portion of the data is stored at the client terminal.

Implementations of the invention may include one or more of the following. The network may be a LANnet, a WANnet, the Internet or an Intranet. Each of the client terminals may be a personal computer, an organizational personal computer, or a stand-alone terminal. The stand-alone terminal may further include hardware to execute the loading of smart cards, the reading of magnetic cards, or the processing of videographics. The servers may include an information server, a transactional server, or an external server. The transactional server may perform a financial transaction or a personal transaction. The network may be described as a four-tier model. The system may also include a graphical user interface to display a portion of the data on the client terminal. The system may also be operable to interact with a plurality of third party applications. The interfacing system may be a request broker system. The system may also include a telecommunications device to connect to the network to access data from the servers. The system may also include an optimization system to

distribute data between the client terminals and the servers. The characteristic data may include a plurality of static or dynamic parameters associated with each of the client terminals, or a usage pattern associated with the client terminals.

In another aspect, the invention is directed towards a method for transferring data over a network having a plurality of client terminals and a plurality of servers. The method includes establishing a data communication path between the client terminals and the servers. A database of characteristic data associated with the client terminals and the servers may be generated. The method further includes dynamically distributing data between the client terminals and the servers as a function of the characteristic data and the database. As a result, a portion of the data resides in the client terminals, and a portion of the data resides at the servers.

Implementations of the invention include the following. The establishing step may further include loading an application onto one of the terminals, and establishing the data communication as a function of specifications of the application.

In another aspect, the invention is directed to a computer-based method of processing information dynamically between a plurality of client terminals and a plurality of servers. The method includes the steps of establishing a data communication path between the client terminals and the servers; generating a database of characteristic data associated with the client terminals and the servers; and dynamically distributing data between the client terminals and the servers as a function of the characteristic data in the database. As a result, a portion of the data resides at the client

terminals and a portion of the data resides at the servers.

Other advantages and features of the present invention will become apparent from the following description, including the drawings and claims.

Brief Description of the Drawing

FIG. 1 is a constructional diagram of an Information Transfer Network System in accordance with the present invention.

FIG. 2 is a block diagram of the components and architecture of the Information Transfer Network System of FIG. 1.

FIG. 3 is a block diagram of the client and the server of the Information Transfer Network System.

FIG. 4 is a constructional diagram of the general architecture of the Information Transfer Network System.

FIG. 5 is a block diagram and flowchart of the processing sequence of a presentation system in accordance with the system of FIG. 1.

FIG. 5A is a block diagram of the object oriented data model used to transfer information between the presentation system and a request broker.

FIG. 5B is a block diagram of a bandwidth optimization system.

FIG. 5C is a block diagram of the request broker data model.

FIG. 6 is a block diagram of the JAVA Applet for the client terminals of the Information Transfer Network System.

FIG. 7 is a flowchart of the server of FIG. 3.

FIGS. 8a and 8b are a block diagram of a database design in accordance with the present invention.

FIG. 9 is a block diagram of a visual data management tool.

FIG. 10 is a table of a database example depicting the structure of a database of the system of FIG. 1.

FIG. 11 is a table of the database example depicting the object link types of the database of FIG. 10.

FIG. 12a is part A of the table of the database example depicting the data in the database of FIG. 11.

FIG. 12b is part B of the table of the database example depicting the data in the database of FIG. 11.

FIG. 12c is part C of the table of the database example depicting the data in the database of FIG. 11.

FIG. 13 is an illustration of a classification tree and associated data items of the database example of FIG. 11.

Description of the Preferred Embodiments

FIG. 1 shows a block diagram of an implementation of an Information Transfer Network System 100. Generally, system 100 provides a medium for presenting visual information (multimedia), performing business transactions, and providing advanced services to customers through publicly accessible terminals or home personal computers connected to a network. The system includes an infrastructure 2. Infrastructure 2 may be a LANnet, WANnet, Internet or Intranet. References to the terms "backbone" and "infrastructure" are to be similarly construed to represent a network for supporting the distribution of information, for example, data to and from a multitude of terminals and servers connected to the network. The system also includes one or more client terminals 1 connected to backbone 2. Client terminals 1 may be stand-alone terminals which are able to process, for example, credit card information or execute video

applications. The system also includes a plurality of personal computers "PCs" 3, an organization network 5, a data management system 6, and a control management system 7. The data management system 6, the control management system 7, the organization network 5 and PCs 3 are connected to infrastructure 2. Data management system 6 manages existing and new entry data tasks. Data management system includes a copy of information located in the database of information server 8. At predefined periods, the database of the information server 8 is updated by the data management system 6. The database of information server 8 is the "formal" active database of system 100.

Organization PC terminals ("OPCs") 4 and stand-alone networks 1' are connected to infrastructure 2 via organizational network 5. Stand-alone terminals 1 and 1', PCs 3 and OPCs 4 may execute similar applications. It should be noted, however, that stand-alone terminals 1 and 1' may include hardware and peripherals which may execute additional applications, for example, loading smart cards, reading magnetic cards, and processing video graphics.

System 100 also includes an information server 8, a transaction server 9, and an external server 10. A content provider server 11 is also shown connected to the network backbone 2 via external server 10. The system 100 may also include thousands of terminals, servers, and network facilities.

Generally, PCs 3 connect to infrastructure 2 in a conventional manner using the hypertext transfer protocol (HTTP). This initiates a request for a secure or non-secure connection to information server 8 via backbone 2. Once the PC is verified and identified, a connection between the information server 8, and the PC 3 is established. Once the connection is established, the

type of application to be executed is determined. The specifications of the application may be loaded onto the terminal from the server or from a hard disk. The specifications are used to determine if a non-secure or secure connection should be used. (See FIG. 6) For example, financial transactions or transactions containing personal information may require a secure connection.

In one operating process, a financial or personal transaction may be requested by one of the terminals. In this regime, the information server 8 connects the application with the transaction server 9 via the information server 8. Transaction server 9 may process the request locally, or may establish a remote connection to a third party clearing system.

In another processing regime, the application may require additional information that is not located on either of the information server 8 or transaction server 9. In such a case, the information may be retrieved from content provider server 11. To receive the necessary information, information server 8 establishes a connection via an external server 10 to a content provider server 11. In this regime, external server 10 establishes a gateway to content provider 11.

Once the type of connection is established, an application channel is formed. The application channel is used to transmit information from the database of server 8 to PC 3 upon the request of the user. The information may be stored or displayed by the user.

FIG. 2 is a more detailed block diagram of the components and architecture of the system 100. The Client 12 includes Smart Decompression Engine 13 connected to Browser 14. Browser 14 includes Java Applet 15. Browser 14 is connected to External Devices 16. External Devices 14 is connected to Control and

Management Agent 17. Client 12 is connected to Secure HTTP Channel 31, which is connected to Server 18. Server 18 comprises Smart Compression Engine 19 connected to a Control and Management Agent 21, which is connected to services and a database. Control and Management system 23 includes Configuration Database 26, Agent Manager 27, and Optimizing Engine 28. Optimizing Engine 28 includes Mathematical model 24 and Optimization Algorithms 25. Data Entry System 29 includes Visual database manipulation tool 30.

FIG. 3 is a detailed block diagram of the Browser 14, Java Applet 15, Secure HTTP Channel 31 and Database 22 components and architecture. Browser 14 comprises Java Virtual Machine 14. Java Applet 15 comprises Application Classes 34. External Devices 16 are connected to Java Applet 15 by Com Bridge 32. External Devices 16 are connected to Control and Management Agent 17 by SNMP 33. Visual Data Management 29 is connected to database 22 by Visual Data Insertion channel 31'. External Services 39 are connected to Services Database 20. Services Database 20 includes Service Broker 35, Java Classes 36, Application Data 37 and Multimedia Data 38.

FIG. 4 shows a constructional diagram of the general system architecture of the Information Transfer Network System 100. Request Broker 40 comprises a HTTP Request Broker 41 connected to Request Broker Interface Modules 42, and a Content and Software Transmitter 43 connected to the Profiles Manager 44. The Request Broker Interface Modules 42 are connected to Clients 12 and Application Services 20. Content and Software Transmitter 43 is connected to Clients 12 and Databases 22. External Services 39 are connected to Application Services 20.

The Information Transfer Network system architecture is based on a unique 4-Tier model architecture system that allows third-party components to be integrated in a plug-and-play fashion. This architecture allows the Information Transfer Network system to use different Graphical User Interface ("GUI") and multimedia interfaces to distribute software and information over the network, and to interact with third party applications and databases. The Information Transfer Network system uses Object Oriented Design (OOD/OOP), a component based system, which permits the re-use and re-purposing of functions.

The first tier, the Presentation system, resides on the client machine. The other tier systems may also reside on the client machine, on a single server, or across multiple servers over the network. Tiers can be duplicated on multiple servers for security, load balancing and optimization of system performance. The separation of these functions into tiers enables the system to adapt to different environments and technology standards. For example, the Presentation system can adapt itself for use by a home PC user (Internet), or specialized terminals.

FIG. 5 shows an implementation of the processing sequence of the presentation system. The presentation system incorporates a thin-client methodology, thereby containing only minimal functions required to display (GUI), and the drivers for essential internal devices (sound card, video card, network card, etc.). All advanced interface features, business logic, and services may be loaded onto the client terminal dynamically on system demand. The information or intelligence at the client terminal varies according to the services demanded by the user. The Presentation system is implemented using Java architecture, and executes inside a standard

web browser that supports a Java virtual Machine. Suitable browsers include the Microsoft Internet Explorer version 3 (IE3) or the Netscape Navigator version 3. Java is a commercially available multi-platform Object Oriented language that adheres to the Client/Server model, and may run on any machine that has a Java Virtual Machine.

Referring to FIG. 4, the Presentation system passes all requests for information or services through the Request Broker system 40. The presentation system may interact with the Request Broker system through the Request Broker Interface using, for example, Java, JDBC, CGI, ActiveX, CORBA, HTTP, Network Pipe or other interface protocols.

Java Applets are small blocks of computer code which may be loaded onto the Client 12 from the Server 18 to be executed by the Client 12. Java Applets may perform interfacing between the Presentation system and the Request Broker 40.

The following are protocols which may be used to perform various functions of the presentation system.

Java Database Connection ("JDBC") is an incarnation of the well- known Open Data Base Connection ("ODBC") protocol that interfaces with different types of databases. The JDBC is the Java flavor of ODBC developed for the Java language.

The Common Gateway Interface ("CGI") may run scripts on the server 18 to translate client requests into calls for information to external programs, applications, or servers.

ActiveX is Microsoft's product based upon Objects interfacing model (OCX) adopted for internet components interfacing. An ActiveX component is a code that may run on the client 12 to handle interfacing with the server 18.

Common Object Request Broker Architecture ("CORBA") allows applications to communicate with each other irrespective of the type and location of the application. CORBA defines an Interface Definition Language (IDL) and the Application Programming Interface that enable client to server object interaction.

Network Pipe is a dedicated protocol used for streaming data. This protocol is used to establish or maintain a continuous flow of data, for example, in a telephone conversation or video clip.

The presentation system is insensitive to the content being presented, since the functions required to display material are downloaded as objects with the material. Multiple GUI templates may be created to offer different displays for the same content without duplicating the other tiers. For example, two different client terminals, such as PC 3 and OPC 4 could display different views of the same content using the same tiers.

Generally, the presentation system will start the application by executing HTML File 50, and will load main Java application 52. The presentation system, using main java applets 52, will present Java screen A 54 to the user. Java screen A 54 comprises Java object-Movie 60, Java object-Button 56, and Java object-Button 56. If Java object-button 56 is activated, the presentation system will load Java Screen B 54'. If Java object-button 58 is activated, the presentation system will load Java Screen C 54".

FIG. 5A shows a block diagram of the data model used to transfer information between the presentation system and the request broker. The Presentation system includes a Graphical User Interface. The GUI Multimedia Components of the Information Transfer Network system include: I a) JAVA-HTML Classes that enable the use of

HTML within Java applets, including rich text format, pages layout, and hyperlinks (between items in Java application); b) controlled WWW Links Java applets that can completely control a standard HTML session. This includes replacing browser toolbars and menus with Java navigation toolbars, and capturing and filtering all hyperlinks; c) peripherals interaction applets. Object 60 comprises various data objects to be presented by GUI to the presentation system.

FIG. 5B shows a block diagram of the bandwidth optimization system. The optimization system includes Bandwidth Optimization Methods for compression and decompression of GUI components and Java Classes. The Client 12 comprises Control and Management Agent 17. Server 18 comprises Control and Management Agent 21. Control and system 23 comprises Configuration Database 26, Agent Manager 27, and Optimizing Engine 28. Optimization Engine 28 includes Mathematical model 24 and Optimization Algorithms 25. Control and Management Agent 23 is connected to the Client 12 and Server 18 using Agent Manager 27. Agent Manager 27 is connected to C&M Agent 17 and C&M Agent 21.

FIG. 5C shows a block diagram of Request Broker data model. Java Screen 90 comprises Java Object-Item List 92 and Java Object-Keyboard 94. Request Broker 40 is connected to database 22 using Database Driver 106. Request Broker 40 is connected to presentation system through Java Screen 90 using Optimization Agent 104 and Translator 102.

The Information Transfer Network system 100 utilizes a Forms Generator/Processor that includes: a) Form Filling system that enables users to complete self-service forms on screen for electronic submission or printing; and b) Validity Checking system. The Forms may

have embedded validity checking, which runs on the client 12.

The Information Transfer Network system 100 utilizes Authoring and Data Management Tools including: a) Application Generator- Rapid Application Development (RAD) tools for building self-service applications; b) Object-Oriented Dynamic Display System that Reads GUI objects from databases on-the-fly, and displays them according to Properties and Methods; and c) Visual Data Management- Visual tools for organizing, classifying, and maintaining data items.

The Information Transfer Network system 100 utilizes Database Technology that includes a Generic Database Design. a Standardized Data Modeling may be used to represent all applications, and GUI objects. This permits search, data entry, and database distribution tools to be operated on every application without modification.

FIG. 6 illustrates a start-up routine for the JAVA Applet for the Information Transfer Network System 100 client terminals.

FIG. 7 shows a flowchart of the processing sequence of the applications system of Server 18.

FIGS. 8a and 8b show a block diagram of the generic database design. The basic concept of the database model is based on three components: OBJECTS, the PROPERTIES of each object, and the LINKS between the objects. The database model contains the META-DATA which represents the structure of the data, and determines the laws of data including OBJECT_TYPES: the types that the objects may be; PROPERTY_TYPES: the types of properties each object may have; DATA_TYPES: the data types (numeric, string, etc.) a property may be; LINK_TYPES: the kinds of links that may be between object types; and

LANGUAGES the different languages that the data may be translated to.

Several architectural and functional components of system 100 will now be described in detail.

Request Broker System

The Request Broker system 40 acts as a link between the Applications system and the Presentation system. The Request Broker system 40 interprets the request from the Presentation system, and locates the appropriate Request Broker Interface server. Suitable servers include Oracle, Lotus Notes, and GIS. Request Broker System 40 passes the request to the server application, receives the results, and returns the results to the Presentation system. Because the Presentation system interacts with the servers only through the Request Broker system 40, the Presentation Layer does not maintain information concerning the location of the loading of the application. A single Request Broker system may serve multiple applications, or a specialized Request Broker system may be implemented for a single application.

The Request Broker system interfaces to the Applications system (Business Logic) layer using specific protocols for each type of server. For example, information applications will use the Oracle Web Server protocol, and Java will be used to connect with servlets (server-side applets).

Request Broker system architecture results in an open environment that can integrate new types of servers. For example, a new third-party electronic commerce server that implements SET security may be integrated into the Request Broker system.

A single Request Broker system may handle multiple, simultaneous requests. This allows traffic to be monitored and controlled, and eliminates the need to run a separate process for each request as would be

required using the CGI protocol, or direct requests from the client terminals to the applications.

Application System (Business Logic System)

The Application system tier contains all of the applications and services to be used by the end-user. This tier may include applications developed specifically for the Information Transfer Network system 100, or applications from third parties, written in any standard language or with any tool.

For example, third-party applications that utilize a world-wide- web interface may interface with the Request Broker system using the World Wide Web Request Broker Interface. Applications that do not utilize a world wide web interface may interface with the CORBA to Java, and Java to CORBA Request Broker Interface modules. Thus, the Information Transfer Network system 100 has the ability to incorporate or integrate third-party applications which may run on separate computers, or even within legacy systems such as mainframes.

Database System

The Information Transfer Network system 100 may use the industry standard Oracle 7.3 RDBMS engine for its applications, and meta data storage. The applications may interact with most standard RDBMS (Informix DB/2 SQL Server) or non-RDBMS (Xbase, IMS). Third-party applications may use any database or combination of databases on any platform.

Databases may be distributed across multiple servers over the network as required to optimize performance. For example, video clips can be considered as part of the same logical database as other content (from the Request Broker system point of view), even though the data may be physically located on a

specialized video server, or possibly on the client terminal machine.

Java Architecture

The architecture of system 100 is based on object oriented design using, for example, Sun Java architecture. Java provides the ability to download objects (classes) as they are required. Suitable development systems include Sun Microsystems JDK 1.1.1, Microsoft J++, Symantec Visual CafJ Pro, Borland Jbuilder and Microsoft SDK 2 for Java, and Java Beans. A single Java Bean component may function as a first-class component within a range of applications, including applications developed in Visual Basic, Visual C++, Word, Excel, and Internet Explorer.

An Active-X bridge Java Bean will enable the Information Transfer System to incorporate a wide range of available Active-X controls, without relying on its proprietary Active-X/Java COM Object solution.

The present invention provides the following Java Packages: a) Dynamic Display Package: which reads the properties and methods of objects to be displayed, and displays them according to these properties; b) Display Package which displays and fully controls an HTML window within a Java page; executes hyperlinks to related content, browses Web sites from within the Java environment using a customized toolbar, and controls and filters links to the WWW; c) Internet Optimization Package compression graphic classes for low-bandwidth users; d) Database Connection and Manipulation Package connects to Oracle DB and other databases, searches and retrieve information, and e) Peripheral Support Package which handles Device drivers to communicate with peripherals.

The system may also include the following: a) forms generator that includes off-line tools to scan, OCR, and create electronic forms, display forms on-screen, real-time filling of forms with basic validity checking; submitting and processing of form results, and printing of forms (blank or completed); b) Advanced Search Engine: Soundex; Thesaurus; Morphology (example: lingual = language); expand search to WWW; c) Virtual Reality Interface which simulate real world reality in an interactive 3-D world (e.g. a virtual supermarket that a user can "shop in".); Browse the 3-D world connects items in the 3-D world to Database items; and d) Advanced multimedia capabilities, which include streamed Video, and Audio-supports rich video and audio; Full motion Animation - high performance 2.5D animations; Spatial Audio for Java- provides 3D sound.

Internet Technologies

Browser

The Information Transfer System thin-client is designed to operate inside any standard web browser with a Java Virtual Machine. The browser may be customized to fully control the GUI. For example, the standard browser buttons, toolbars, and menus may be replaced with a multimedia GUI that is suitable for mouse or touchscreen use. Using Java Classes, help functions, sounds, and other multimedia capabilities may also be added to the interface.

Software running within the browser environment communicates with the hardware through a Component Object Model Object (COM Object) that controls device drivers (e.g. card readers, coin machines, laserdiscs, and video cameras). The COM object can interface with and activate Microsoft Active-X controls, such as NetMeeting Video Conferencing. This integration of Java and Active-X may

be facilitated through the Java Active-X Bridge Java Bean, rather than through the COM object. The thin-client may be extended and enhanced using third-party browser plug-ins. This is particularly beneficial for providing enhanced multimedia effects for the GUI, such as Virtual Reality Modeling Language (VRML) for 3-D interfaces.

HTML Manipulation

The Information Network System client integrates the Java Classes, which support HTML 3.2 documents for display of Rich Text Format (fonts, colors, point size) and images on-screen. The HTML document may contain hyperlinks to related information items within the application, or to related sites on the World Wide Web ("WWW"). Upon linking to a WWW site, the screen displays the site in a frame on the upper portion of the screen. The Java-based navigation GUI panel interface is displayed on the lower portion of the screen. The GUI provides the essential functions for Web Browsing, with all browser toolbars and menus removed from the screen. This feature is achieved by first loading a special HTML page which consists of two frames. The lower frame displays a Java applet containing a multimedia GUI. The lower frame controls the upper frame, which contains the linked HTML page. This is implemented completely in Java, and does not rely on a browser or platform specific API. WWW links may be filtered in order to control or limit access to specific sites. An on-screen keyboard may be used to enter specific URLs through a touchscreen.

Optimization of Network Traffic

Real-time compression techniques may be implemented to move information and applets from the server 18 to the client 12. Upon receiving a request,

the server determines what type of network connection exists. A fast LAN connection may be sent full quality, uncompressed files including text, images, video, and Java Classes. For slower WAN connections, the server may compress the files in real-time, which may then be transmitted over the network and decompressed at runtime by the client. Each type of information item has a unique compression method, which is most suitable for the specific item. For example, JPEG compression is used for images, and CAB or JAR compression is used for Java Classes. The JAR technology is also capable of compressing data information and graphics in addition to the Java classes. For images, compression ratios of 1:10 may be achieved using JPEG.

Channel Technology (push)

One standard for accessing information through the Internet is known as pulling, whereby users request specific sites through a Web browser. The sites are then viewed in real-time, as they are loaded. Another standard for accessing information on the Internet is Push technology, which sends information to a client terminal automatically. Information may be general purpose content, or specific content based on a profile or set of parameters established in advance by/for the user. Content is delivered as different channels. Each channel offers different types of content, and the user may specify how often the channel should be updated. For example, a user could subscribe to a newspaper channel in order to receive a customized daily (or more frequent) newspaper based on his/her specific topics of interest.

Integrated Internet Applications

The Information Transfer Network 100 may integrate standard Internet-based applications. For example, e-

mail clients, newsgroup readers, chat rooms, and access to proprietary online services (AOL, CompuServe, MSN, etc.) may be integrated into specific applications. The integration may be performed by Active-X to Java COM bridge, or through a CORBA server to activate remote objects in third-party applications. For example, an application may offer an electronic suggestion box, which will forward suggestions via e-mail to the intended recipient. The HTML interface allows users to search the Internet within a controlled and filtered environment.

Firewall and Security

The Information Transfer Network system 100 integrates a "firewall" to secure servers and systems residing on the Internet infrastructure. This allows the system terminals to be networked through the Internet, or through private Intranets. When connected through an Intranet, the terminals may still offer access to the Internet via a firewall protected gateway at the server. The firewall encrypts data between the client and server when networked through the Internet, and controls access in and out of the system. A SSL secure channel method may be used to provide additional security for transaction data, for example, when performing a financial transaction with server 9.

Network Computer (NC)

The Java-based system may be executed on NC machines being developed by, for example, Sun, Oracle, and IBM. The system may be downloaded to the terminal and run under the Java OS.

Advanced Multimedia and Graphics Technologies

The Information Transfer Network System 100 incorporates full multimedia and graphics capabilities.

Video and audio may be streamed when delivered from the server to the client. This allows the user to view the video concurrently, while it is being downloaded.

The Information Transfer Network System 100 may also show video clips as part of the Interface (in MOV or MPEG format using H.263 standard for video and G723 format for audio. Video Conferencing (2-Way Video) may be employed, such as video conferencing, including white board, document sharing, audio-only Conferencing, and group Conferencing. Also, video cameras may be used to identify users at a terminal, and as part of applications that may require a photo such as a Drivers License application. Other applications include Rich Text Format and HTML and virtual Reality Modeling Language (VRML) Interface in which 3-D interfaces can emulate real-word situations. For example, a VRML supermarket, which allows a user to stroll the aisles, and place items in a 3-D virtual shopping cart, may be employed. Cosmo VRML plug-in for displaying 3-D virtual world interfaces may also be employed. Applications may relate from 3-D items in the virtual world to information items in a database. In the example above, a user could select a grocery item in a 3-D interface, and retrieve nutrition information from the database. Geographic Information Systems (GIS) which include vector and raster maps of the entire United States and other countries may be implemented. Any address (such as ATM locations) can be geocoded and plotted on a map, and a routing system can produce point-to-point driving directions to provide transit maps and point-to-point directions.

Advanced Services and Applications

Forms Generator/Processor

The forms generator scans paper-based forms and converts them to HTML documents using OCR technology.

Form blanks are automatically converted to fields which may accept data. Java Applet may be used to process the forms from the form generator. Validation functions may be added to each field to perform data checking and validation at the client. For example, an account number may be verified by the form processor using a check-digit. Blank or completed forms may be outputted to a printer or form input may be submitted to an application, or a database on the server.

Computer-Telephone Integration (CTI)

Server-based software modules enable users at a terminal to connect into a public switched telephone network to access information for example, a hotel location. The CTI applications are implemented using Microsoft standard Telephone API (TAPI), running on a Windows NT server with a Dialogic voice/modem card or any other voice/modem card which is Microsoft TAPI compliant. The server side of the application is developed using JTAPI. A two-way CTI application, where an application can interact with a person on the other line, is also possible. In this type of application, the end-user would not speak directly to the other end-user. Instead, a server may "speak" using text-to-speech synthesis (DSP), and receive responses using touch-tones (DTMF). The responses may then be converted to messages for display on the end-user's screen, and the session may continue. The system allows users to connect with businesses (e.g. small hotels) that are not computerized, while still allowing the content provider server 11 to control the transaction. The system may also allow communication to take place across language barriers.

Multilingual Capabilities

The Information Transfer Network system 100 supports multilingual audio and bi-directional text display, including Latin and non-Latin characters. Languages may be switched on demand, for example, between English, Spanish, French, German and Hebrew. Multilingual technology through special Unicode character sets may be controlled by Java Classes on Windows-based on non-Windows based machines. The Dynamic Presentation Generator (described in detail below) provides for multiple languages within an application.

Authoring and Data Entry Tools

Rapid Application Development (RAD) contains a set of tools that enable non-programmers to design, build and administrate complete applications, which include GUI, interactive features, data modeling, and content management. Each application contains META data to describe the application's structure. Therefore, every application data model and GUI is modeled as a set of objects that have properties (attributes), methods (actions that are made during the user interaction), and links to other objects to which they relate.

When an application is running, all the objects which contain the current GUI interaction are loaded dynamically to the presentation system, and display themselves according to their own display methods. When a user executes a function, e.g. pressing a button, the appropriate Object Method is activated, which may then activate another object (e.g. display a screen, play a video, run search, etc.)

An application may be loaded using the following steps. A graphics template (buttons, backgrounds, logos, etc.) is chosen from a library. The data Objects, Properties, and Actions that implement the application

GUI and user interactions are defined. For example, when the application uses buttons, one has to define the buttons' shape, color, x-y coordinates, and actions that are executed when the button is pressed. The related objects are then linked, and the resulting application model is stored in the generic database model.

During operation (runtime), the dynamic presentation generator reads all the GUI objects of the application from the database model for display. One routine, Reflection (during runtime), allows the application to identify every object type (loaded dynamically), retrieves information about the object's capabilities, and links to other objects. Another routine, Transparent Canvas, allows new objects to be displayed without interfering with background objects or currently displayed objects.

The data entry tool enables insertion and management of content and classification data that relates to the application. The data entry tool visually browses the classifications and data. Once the database updates are inserted, the system's distribution tools distributes all the information to the appropriate databases, according to the optimization algorithm.

Database Technologies

The Information Transfer Network 100 allows application data and GUI to be described as a set of Objects that have Properties (attributes), Methods (actions), and links (connections to other objects). A unique generic database model may model, store, and manipulate the Objects, Methods, Properties, and object links of any type of application data model. The generic model also stores Meta Data, which is used to display and interact with the GUI of the application.

Example of using the Generic Model to implement a Company's data system

FIGS. 10, 11, 12a-c and 13 illustrate an exemplary application using system 100. Two main classifications are shown: Computers and Food. Within each classification, there are associated companies: G, M, and I. Each Company has products such as Multimedia Stations and Terminals.

Table 1 of FIG. 10 and Table 2 of FIG. 11 show the "META data" which describes the Object Types, Properties, and Link Types. Table 3 of FIGS. 12a - 12c show sample data filled into the model established in Tables 1 and 2. FIG. 13 is an illustration of the classification tree and associated data items of the example.

This model permits many different applications to share a common database design. Therefore, the same search and statistical analysis tools may be used within any application without modification. Thus, the database design, search, data entry, and statistics tools are independent of the content being represented.

Dynamic Distribution

Application databases (which contain information, pictures, video, etc.) may be distributed and replicated across the network to optimize overall system performance. Due to the frequently changing environment in application and information demand, communications bandwidth, and the facilities in the station, information may be redistributed across the network. For example, if there is high demand for a certain video clip, the clip may be distributed for storage on a server near the terminal, or even on the terminal hard drive.

The decision of where and how to distribute databases is made by the system. The decision is a function of all system parameters. The system also

determines the optimal distribution of information to achieve the most efficient performance. The application is not modified upon changes in the database distribution.

The following describes the process and architecture of the dynamic distribution and system optimization systems.

Statistical Analysis

Every operation (menu choices, coordinates of screen touches) within the system is recorded in a database table. This data may be analyzed using statistical analysis tools, including Oracle tools, in order to determine usage patterns, and to improve or alter system performance. Usage patterns, such as printer usage may be analyzed to predict, for example, when the printer will run out of paper. The number of screen touches of a component of the GUI may also be monitored.

For each user operation, and some of the system's activities, a text entry is added to a log file which maintains a statistical record. Each entry line of the log file records the time and date stamp of the action, the type of action, and some general system status information the status information may be used to decode the action recorded in relation to the system's operation status. The log file is transmitted to the control and management system on predefined schedules or on a specific request from the control and management system. The log file may then be analyzed to determine the needs of the system including maintenance and correction. Other types of information may also be retrieved from the log file, including specific application's parts usage and user interface usage.

Systems Management Tools

The Information Transfer Network system 100 is a large, complex, changing, non-deterministic network, handling massive amounts of information, transactions, and services. To manage this network, an Optimized Integrated System Management Architecture (OISMA) may be used. The OISMA may perform the following functions: a) consolidate a central console that monitors all system components; b) distribute regional and functional specific consoles; c) determine various system alerts of system B, (pager, fax, sound, send messages to sub-contractor system); d) perform automatic recovery actions to recover from problems; e) takeover any station and execute commands from a remote source; f) track and analyze problems on system using help desk tools; g) track and manage the flow of system transactions; h) monitor usage patterns and statistics; i) record automatically inventory related to network inventory autodetect and control B plug-and-play installation of components; j) control hardware/Software version and configuration; k) perform preventive Maintenance Management and l) maintain Central backup.

Systems Management Technologies

The following are management tools which may be incorporated into system 100.

Simple Network Management Protocol (SNMP) - Framework of OISMA, enables agents across the network to communicate with central management tools.

Integration of management tools into a four tier Architecture: Each tier will have agents that receive orders, collect data, and report back to the management system.

Integration of third-party SNMP agents: Agents for specific software (e.g. Oracle Database), hardware devices, or operating systems.

Software and Information Deployment and Synchronization: Oracle tools, SMS, and other tools for distribution and synchronization.

HP Open View Network Management Framework - GS intends to integrate this set of network and hardware management tools.

Network Control and Management

Commercially available control and management systems, such as the Trivoli system from IBM and the SMS system from Microsoft, may be used in the information transfer network 100.

System Optimization

System Optimization provides the optimal service on all terminals over time. Services include fast response time, availability of services and peripherals (printer, network, etc.), and appropriate adaptation to terminal capabilities (e.g. home PC with low-bandwidth, machine without video capability, etc.)

Optimization may be achieved by one or more of the following. Static parameters of the hardware and peripheral on any of the terminals may be measured. For example, the type of memory, the type of terminal, or the type of network connection to the terminal may be measured. Dynamic parameters including the current available bandwidth, server utilization, the location and nature of the data, and the location of the selective application or service may be monitored. The system may track the number of uses of a particular application or service over time.

Using the above techniques, the system determines the optimal message to fulfill user request. Further, this optimization may be performed at preset schedules during operation to redistribute applications and content across the network.

The present invention has been described in terms of number of embodiments. The invention, however, is not limited to the embodiments depicted and described.

What is claimed is:

1. A system for transferring data over a network comprising:

a plurality of client terminals connected to the network;

a plurality of servers connected to the network;

an operations management system connected to said network for monitoring and storing characteristic data about said client terminals and said servers; and

an interfacing system connected to said network for dynamically distributing said data between said servers and said terminals as a function of said characteristic data, such that a portion of said data is stored at said client terminals.

2. The system of claim 1, wherein said network is one of a LANnet, a WANnet, the Internet and an Intranet.

3. The system of claim 1, wherein each of said client terminals are one of a personal computer, an organization personal computer, and a stand-alone terminal.

4. The system of claim 3, wherein said stand-alone terminals further comprise hardware for executing one of loading smart cards, reading magnetic cards, and processing videographics.

5. The system of claim 1, wherein each of said servers are one of an information server, a transactional server, and an external server.

6. The system of claim 5, wherein said transactional server is operable to perform one of a financial transaction and a personal transaction.

7. The system of claim 1, wherein said network is a four tier model.

8. The system of claim 1, wherein said network further comprises a graphical user interface for displaying a portion of said data on said client terminals.

9. The system of claim 1, wherein the system is operable to interact with a plurality of third party applications.

10. The system of claim 1, wherein said interfacing system is a request broker system.

11. The system of claim 1, further comprising a telecommunications device connected to said network for accessing said data from said servers.

12. The system of claim 1, further comprising an optimization system for distributing data between said client terminals and said servers.

13. The system of claim 1, wherein said characteristic data is one of a plurality of static or dynamic parameters associated with each of said client terminals and a usage pattern associated with said terminals.

14. A method for transferring data over a network comprising:

providing a plurality of client terminals;

providing a plurality of servers;

establishing a data communication between said client terminals and said servers; and

generating a database of characteristic data associated with said clients terminals and servers; and

dynamically distributing said data between said client terminals and said servers as a function of said characteristic data in said database such that a portion of said data resides at said client terminals and a portion of said data resides at said servers.

15. The method of claim 14, further comprising providing said client terminals with hardware for one of loading smart cards, reading magnetic cards, and processing videographics.

16. The method of claim 14, wherein each of said client terminals are one of a personal computer, an organizational computer, and a stand-alone terminal.

17. The method of claim 14, further comprising providing each of said servers as one of a transactional server, an information server, and an external server.

18. The method of claim 17, further comprising performing one of a financial transaction and a personal transaction.

19. The method of claim 14, further comprising providing a graphical user interface to display data on said client terminals.

20. The method of claim 14, wherein the network is one of a WANnet, LANnet, Internet, and an Intranet.

21. The method of claim 14, wherein said establishing step further comprises loading an application onto one of said terminals and establishing said data communication as a function of specifications of said application.

22. The method of claim 14, further comprising a telecommunications device connected to said network for accessing data from said servers.

23. The method of claim 14, further comprising providing an optimization system for distributing said data between said client terminals and said servers.

24. The method of claim 14, wherein said characteristic data is one of a plurality of static or dynamic parameters associated with each of said client terminals and a usage pattern associated with said terminals.

25. The method of claim 14, wherein said network is a four-tier model.

26. A computer-based method of processing information dynamically between a plurality of client terminals and a plurality of servers comprising:

establishing a data communication path between said client terminals and said servers;

generating a database of characteristic data associated with said clients terminals and servers; and

dynamically distributing said data between said client terminals and said servers as a function of said characteristic data in said database such that a portion of said data resides at said client terminals and a portion of said data resides at said servers.

27. The method of claim 26, further comprising providing said client terminals with hardware for one of loading smart cards, reading magnetic cards, and processing videographics.

28. The method of claim 26, wherein each of said terminals are one of a personal computer, an organizational computer, and a stand-alone terminal.

29. The method of claim 26, further comprising providing each of said servers as one of a transactional server, an information server, and an external server.

30. The method of claim 29, further comprising performing one of a financial transaction and a personal transaction.

31. The method of claim 26, further comprising providing a graphical user interface to display data on said client terminals.

32. The method of claim 26, wherein the network is one of a WANnet, LANnet, Internet, and an Intranet.

33. The method of claim 26, wherein said establishing step further comprises loading an application onto one of said terminals and establishing said data communication as a function of specifications of said application.

34. The method of claim 26, further comprising a telecommunications device connected to said network for accessing data from said servers.

35. The method of claim 26, further comprising providing an optimization system for distributing said data between said client terminals and said servers.

36. The method of claim 26, wherein said characteristic data is one of a plurality of static or dynamic parameters associated with each of said client terminals and a usage pattern associated with said terminals.

37. The method of claim 26, wherein said network is a four-tier model.

38. A computer program residing in a computer-readable medium, comprising instructions for causing a computer system to:

establish a data communication path between a plurality of client terminals and a plurality of servers;

generate a database of characteristic data associated with said client terminals and said servers; and

dynamically distribute data between said client terminals and said servers as a function of said characteristic information in said database such that a portion of said distributed data resides at the client terminal and a portion of said distributed resides at said servers.

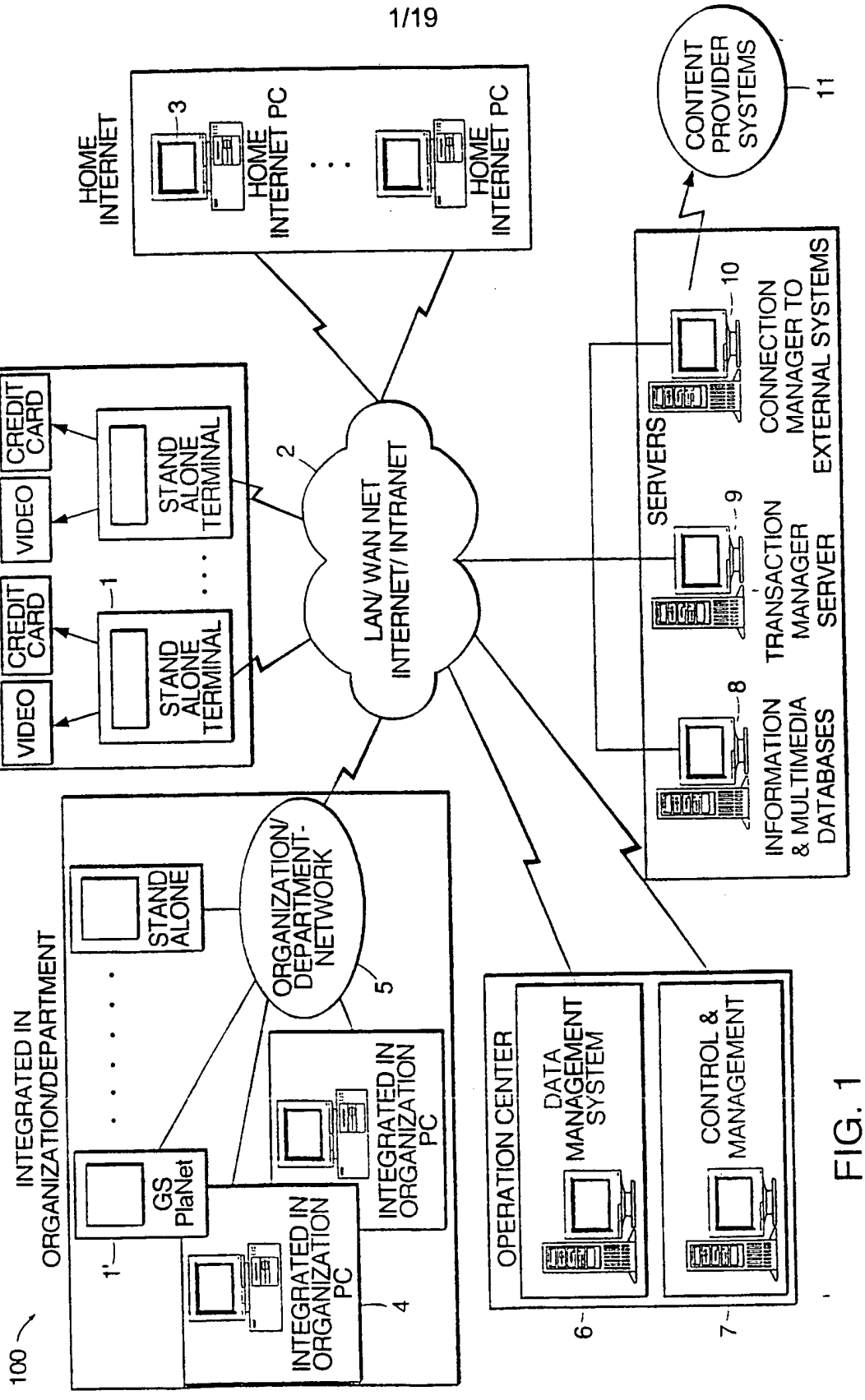

HOME INTERNET
HOME INTERNET PC
HOME INTERNET PC
3
CONTENT PROVIDER SYSTEMS
11
STAND ALONE TERMINALS
VIDEO
CREDIT CARD
STAND ALONE TERMINAL
VIDEO
CREDIT CARD
STAND ALONE TERMINAL
1
2
LAN/ WAN NET INTERNET/ INTRANET
SERVERS
10
9
8
CONNECTION MANAGER TO EXTERNAL SYSTEMS
TRANSACTION MANAGER SERVER
INFORMATION & MULTIMEDIA DATABASES
INTEGRATED IN ORGANIZATION/DEPARTMENT
STAND ALONE
ORGANIZATION/ DEPARTMENT-NETWORK
5
GS PlaNet
1'
INTEGRATED IN ORGANIZATION PC
INTEGRATED IN ORGANIZATION PC
4
100
OPERATION CENTER
DATA MANAGEMENT SYSTEM
CONTROL & MANAGEMENT
6
7

FIG. 1

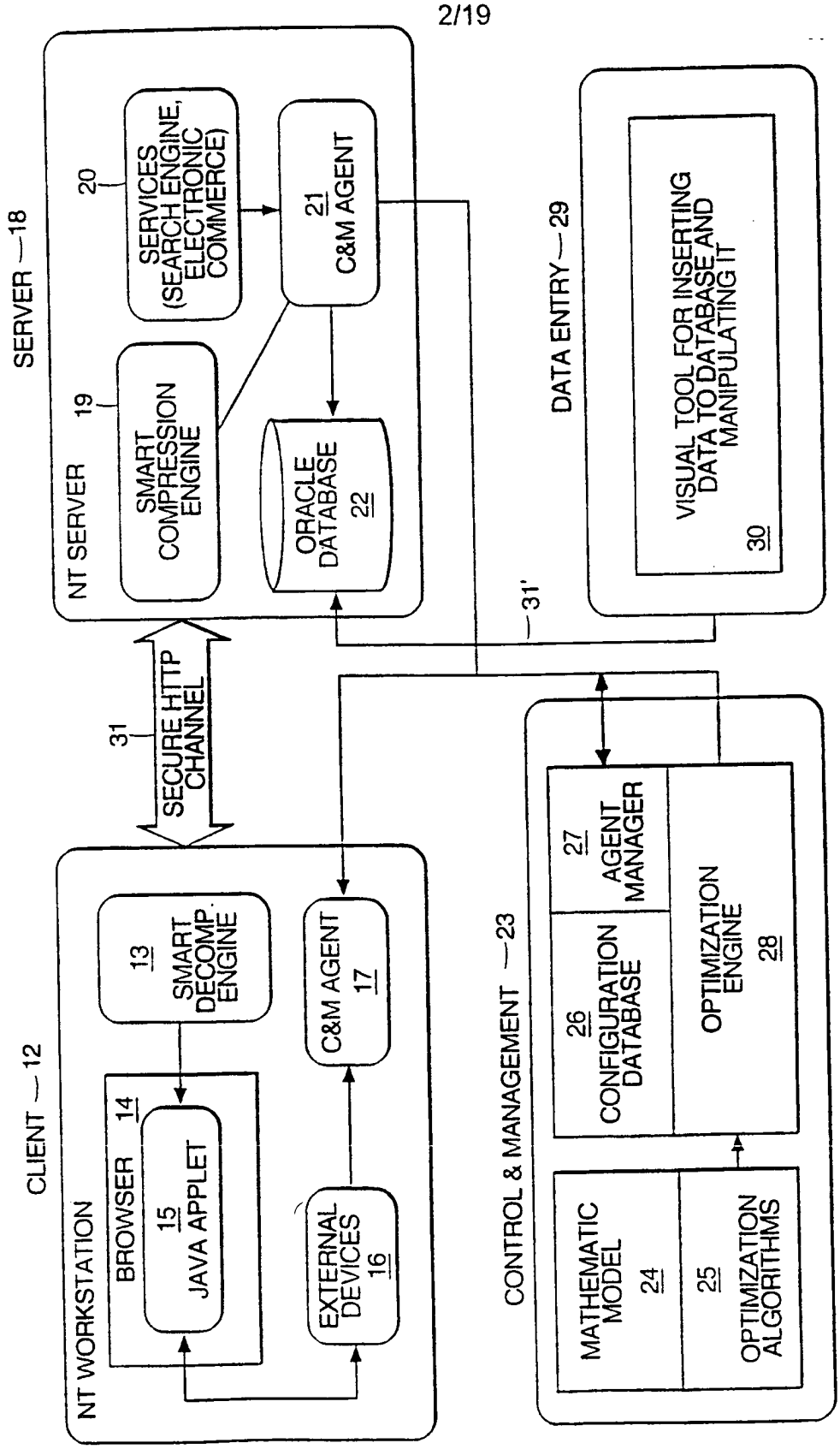

SERVER 18
NT SERVER 19
SERVICES (SEARCH ENGINE, ELECTRONIC COMMERCE) 20
C&M AGENT 21
SMART COMPRESSION ENGINE
ORACLE DATABASE 22
DATA ENTRY 29
VISUAL TOOL FOR INSERTING DATA TO DATABASE AND MANIPULATING IT 30
31'
SECURE HTTP CHANNEL 31
CLIENT 12
NT WORKSTATION
BROWSER 14
JAVA APPLET 15
SMART DECOMP. ENGINE 13
C&M AGENT 17
EXTERNAL DEVICES 16
CONTROL & MANAGEMENT 23
AGENT MANAGER 27
CONFIGURATION DATABASE 26
OPTIMIZATION ENGINE 28
MATHEMATIC MODEL 24
OPTIMIZATION ALGORITHMS 25

FIG. 2

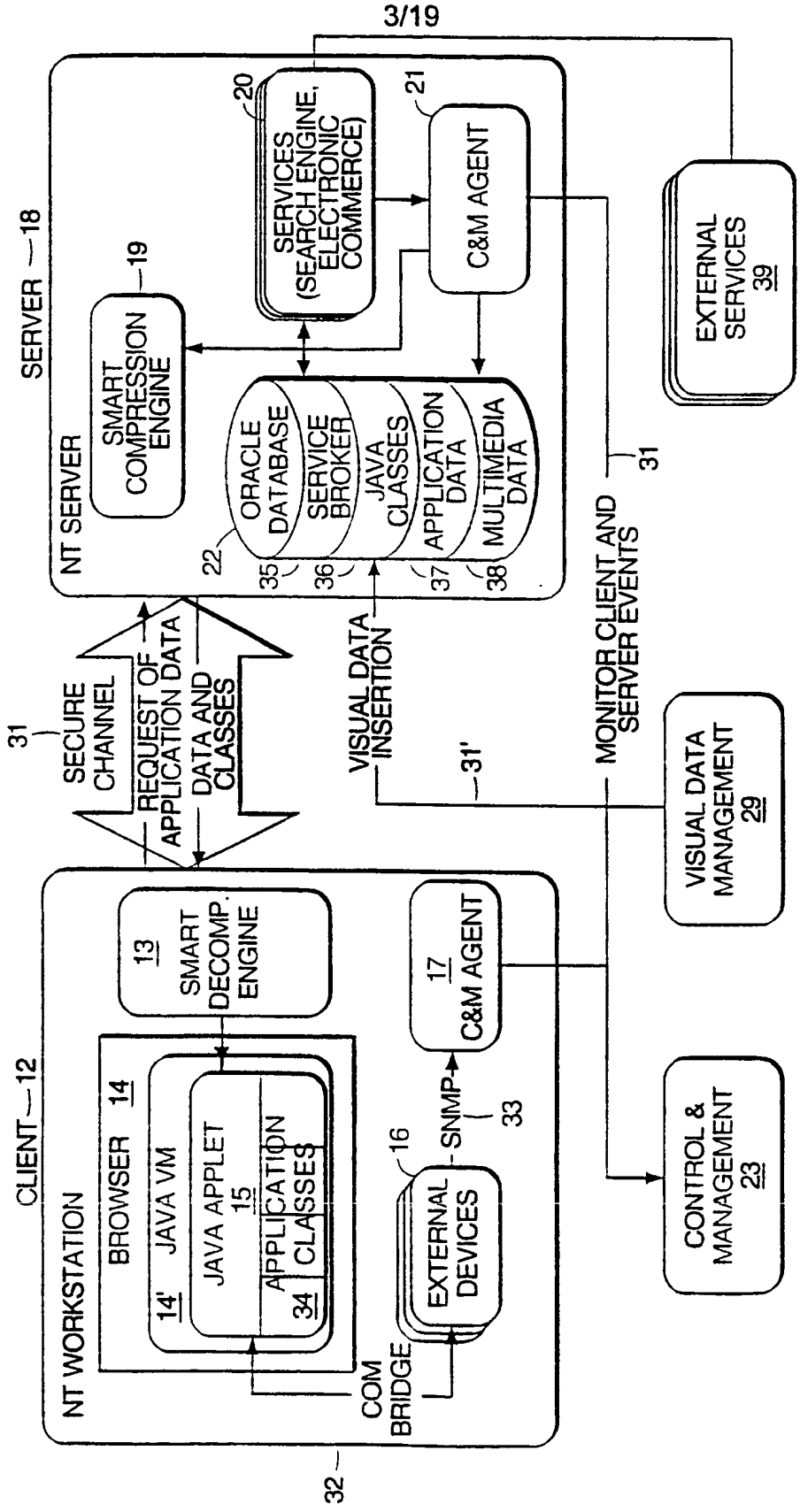

SERVER 18
NT SERVER
SMART COMPRESSION ENGINE
19
SERVICES (SEARCH ENGINE, ELECTRONIC COMMERCE)
20
C&M AGENT
21
EXTERNAL SERVICES 39
ORACLE DATABASE
SERVICE BROKER
JAVA CLASSES
APPLICATION DATA
MULTIMEDIA DATA
22
35
36
37
38
31
MONITOR CLIENT AND SERVER EVENTS
VISUAL DATA INSERTION
31'
SECURE CHANNEL
REQUEST OF APPLICATION DATA
DATA AND CLASSES
VISUAL DATA MANAGEMENT 29
CLIENT 12
NT WORKSTATION
BROWSER 14
JAVA VM
14'
JAVA APPLET 15
APPLICATION CLASSES 34
13 SMART DECOMP. ENGINE
17 C&M AGENT
SNMP
33
16
EXTERNAL DEVICES
COM BRIDGE
CONTROL & MANAGEMENT 23
32

FIG. 3

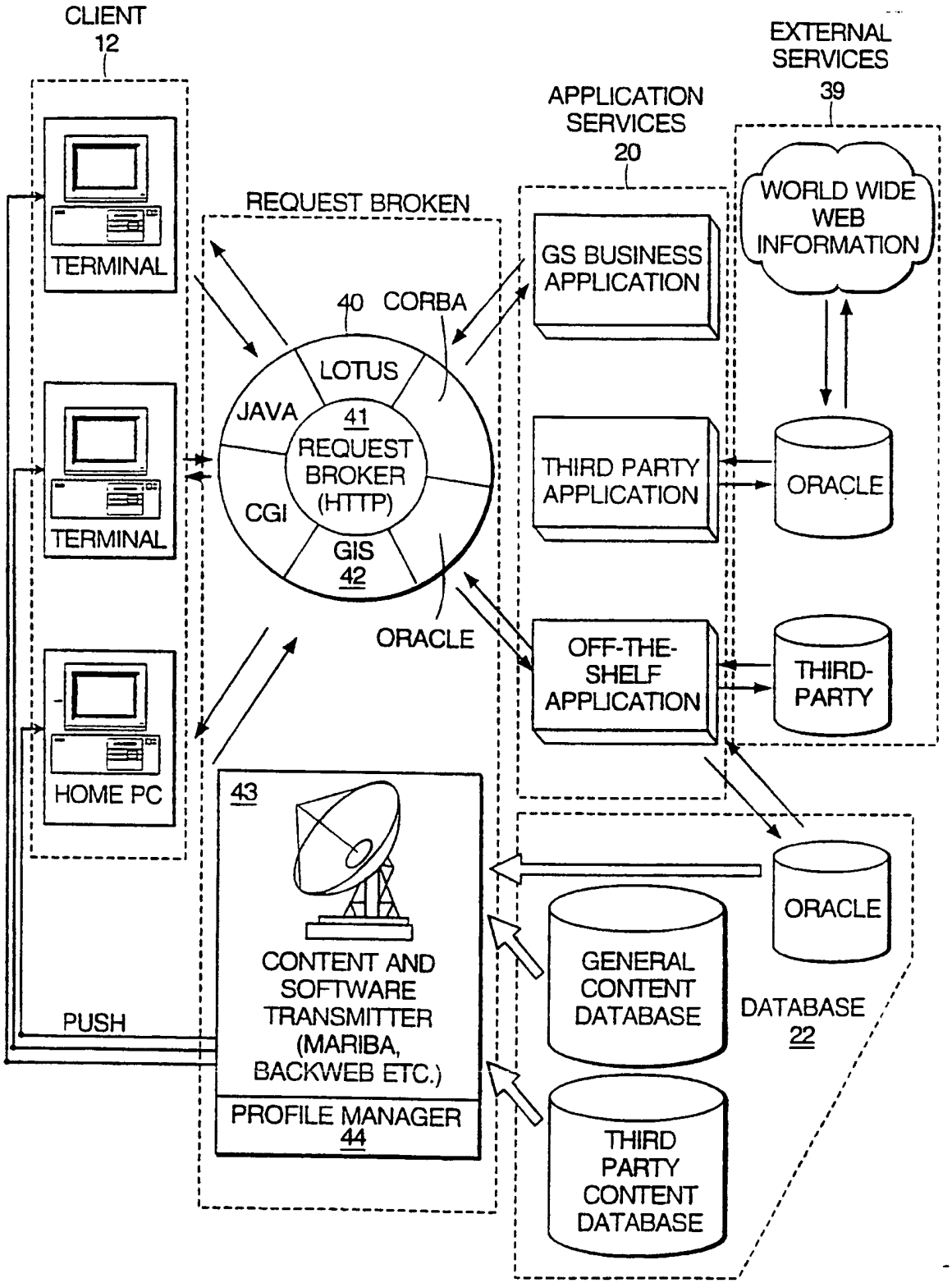

CLIENT
12
TERMINAL
TERMINAL
HOME PC
PUSH
REQUEST BROKEN
40 CORBA
LOTUS
JAVA
41
REQUEST BROKER (HTTP)
CGI
GIS
42
ORACLE
43
CONTENT AND SOFTWARE TRANSMITTER (MARIBA, BACKWEB ETC.)
PROFILE MANAGER
44
APPLICATION SERVICES
20
GS BUSINESS APPLICATION
THIRD PARTY APPLICATION
OFF-THE-SHELF APPLICATION
EXTERNAL SERVICES
39
WORLD WIDE WEB INFORMATION
ORACLE
THIRD-PARTY
ORACLE
GENERAL CONTENT DATABASE
THIRD PARTY CONTENT DATABASE
DATABASE
22

FIG. 4

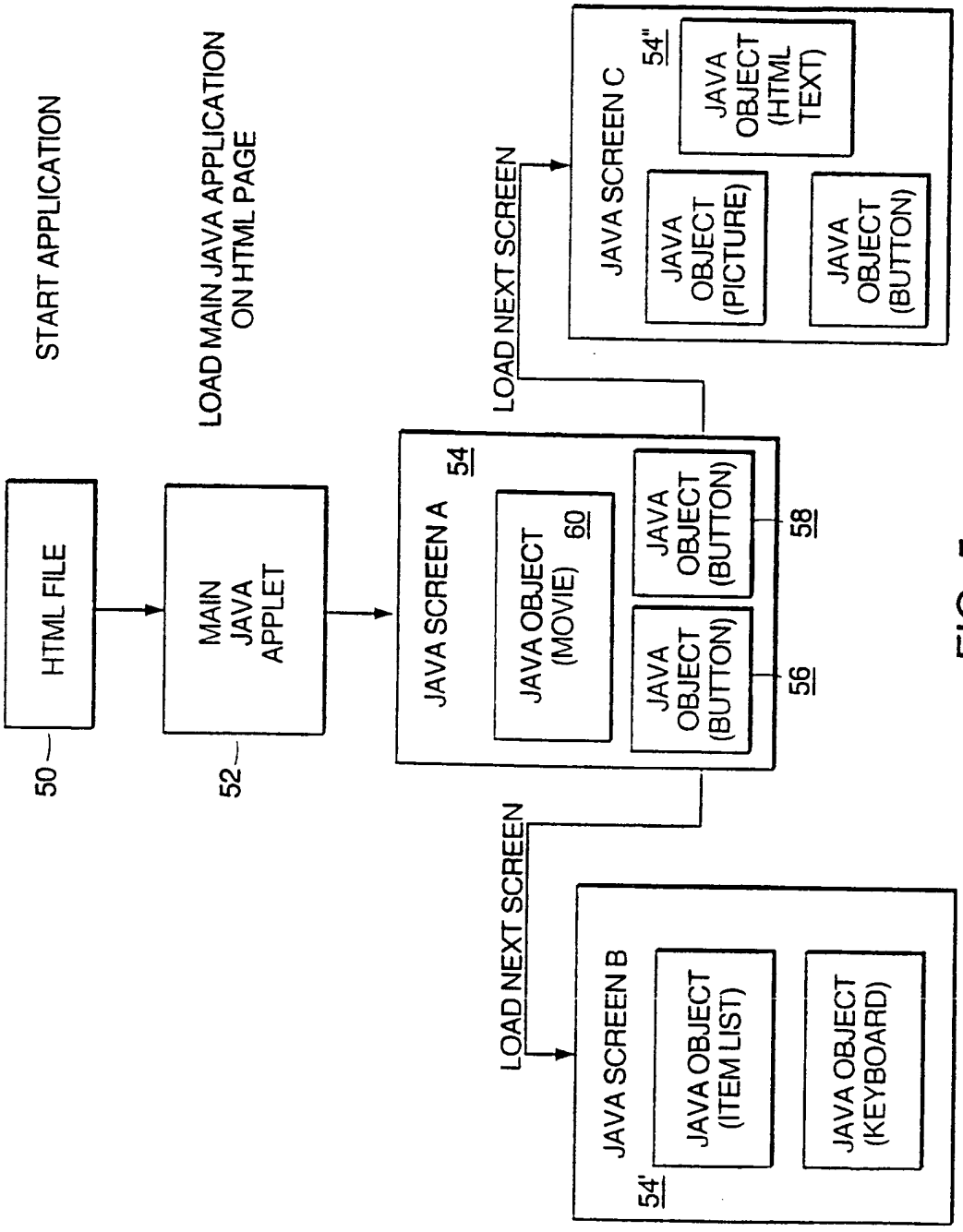
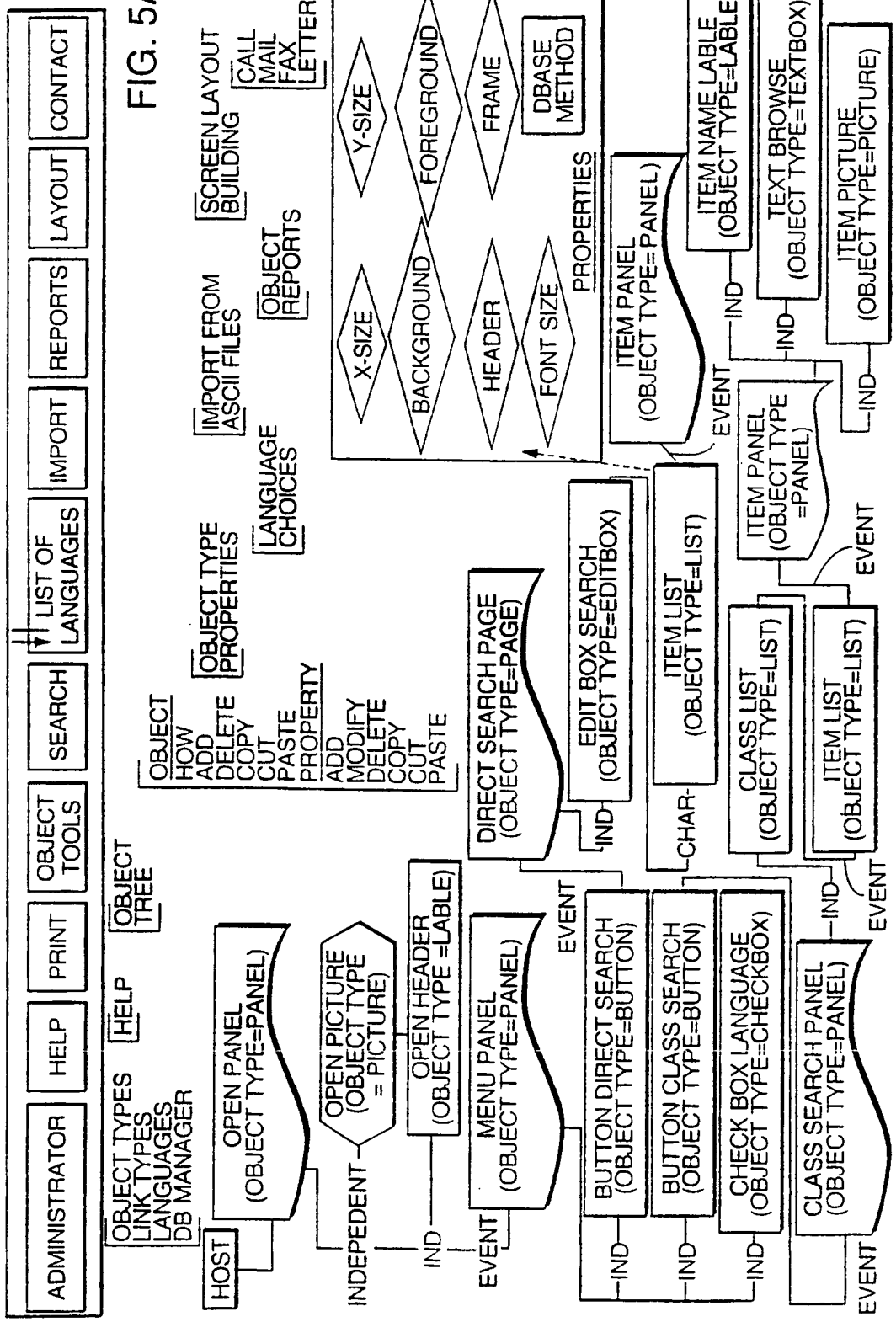

START APPLICATION
LOAD MAIN JAVA APPLICATION ON HTML PAGE
HTML FILE
MAIN JAVA APPLET
JAVA SCREEN A
54
JAVA OBJECT (MOVIE)
60
JAVA OBJECT (BUTTON)
58
JAVA OBJECT (BUTTON)
56
LOAD NEXT SCREEN
JAVA SCREEN C
54"
JAVA OBJECT (HTML TEXT)
JAVA OBJECT (PICTURE)
JAVA OBJECT (BUTTON)
LOAD NEXT SCREEN
JAVA SCREEN B
54'
JAVA OBJECT (ITEM LIST)
JAVA OBJECT (KEYBOARD)
50
52

FIG. 5

FIG. 5A


60
ADMINISTRATOR
HELP
PRINT
OBJECT TOOLS
SEARCH
LIST OF LANGUAGES
IMPORT
REPORTS
LAYOUT
CONTACT
OBJECT TYPES
LINK TYPES
LANGUAGES
DB MANAGER
HELP
OBJECT TREE
HOST
OPEN PANEL (OBJECT TYPE=PANEL)
INDEPEDENT
OPEN PICTURE (OBJECT TYPE = PICTURE)
IND
OPEN HEADER (OBJECT TYPE =LABLE)
EVENT
MENU PANEL (OBJECT TYPE=PANEL)
OBJECT
HOW
ADD
DELETE
COPY
CUT
PASTE
PROPERTY
ADD
MODIFY
DELETE
COPY
CUT
PASTE
OBJECT PROPERTIES
OBJECT TYPE PROPERTIES
LANGUAGE CHOICES
IMPORT FROM ASCII FILES
OBJECT REPORTS
SCREEN LAYOUT BUILDING
CALL
MAIL
FAX
LETTER
X-SIZE
BACKGROUND
HEADER
FONT SIZE
Y-SIZE
FOREGROUND
FRAME
DBASE METHOD
PROPERTIES
DIRECT SEARCH PAGE (OBJECT TYPE=PAGE)
EVENT
IND
EDIT BOX SEARCH (OBJECT TYPE=EDITBOX)
CHAR
ITEM LIST (OBJECT TYPE=LIST)
ITEM PANEL (OBJECT TYPE =PANEL)
EVENT
ITEM PANEL (OBJECT TYPE=PANEL)
EVENT
IND
ITEM NAME LABLE (OBJECT TYPE=LABLE)
IND
TEXT BROWSE (OBJECT TYPE=TEXTBOX)
IND
ITEM PICTURE (OBJECT TYPE=PICTURE)
IND
BUTTON DIRECT SEARCH (OBJECT TYPE=BUTTON)
IND
BUTTON CLASS SEARCH (OBJECT TYPE=BUTTON)
IND
CHECK BOX LANGUAGE (OBJECT TYPE=CHECKBOX)
IND
CLASS LIST (OBJECT TYPE=LIST)
IND
ITEM LIST (OBJECT TYPE=LIST)
EVENT
CLASS SEARCH PANEL (OBJECT TYPE=PANEL)
EVENT

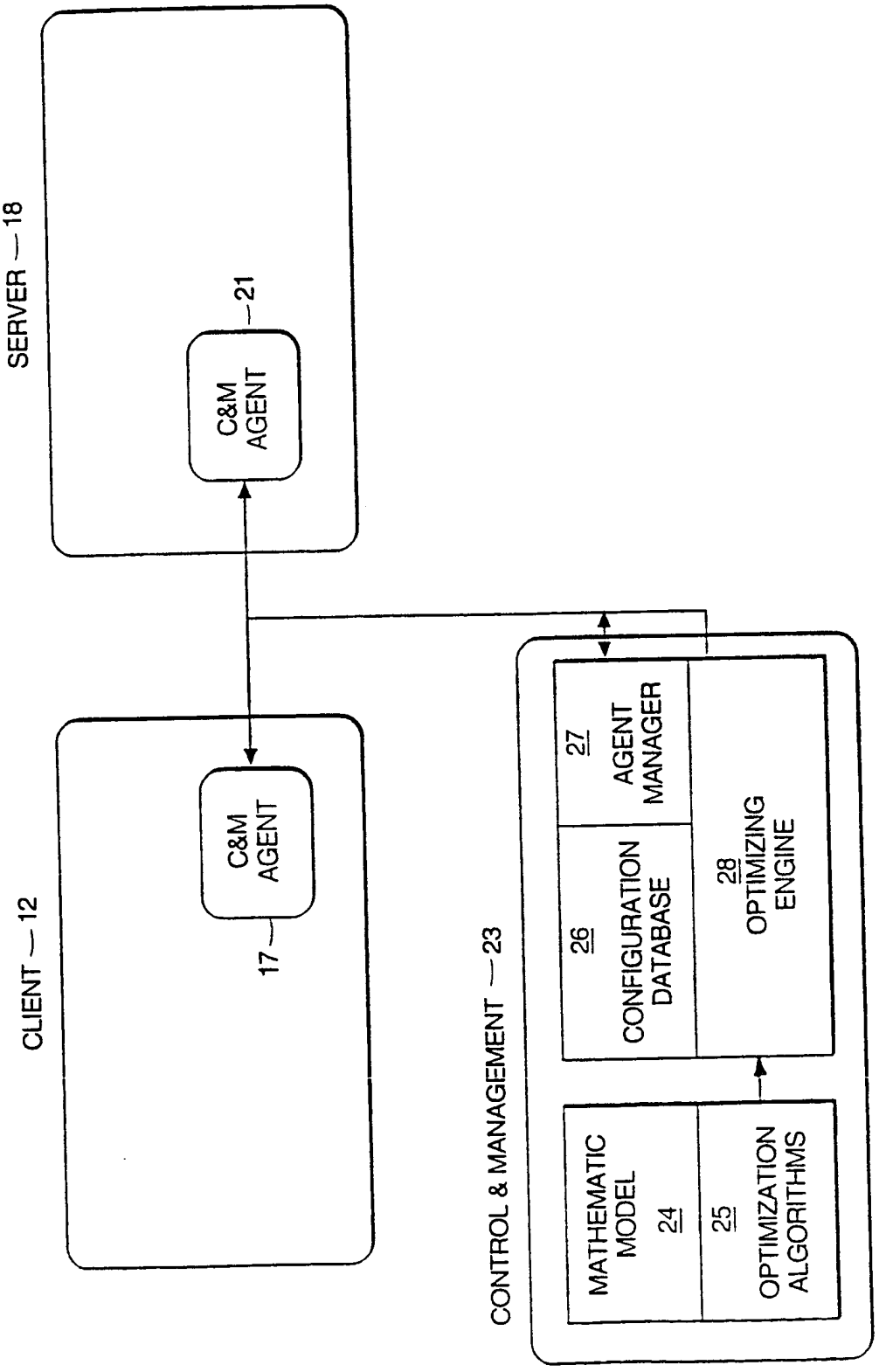

SERVER — 18
21
C&M AGENT
CLIENT — 12
C&M AGENT
17
CONTROL & MANAGEMENT — 23
27
AGENT MANAGER
26
CONFIGURATION DATABASE
28
OPTIMIZING ENGINE
MATHEMATIC MODEL
24
25
OPTIMIZATION ALGORITHMS

FIG. 5B

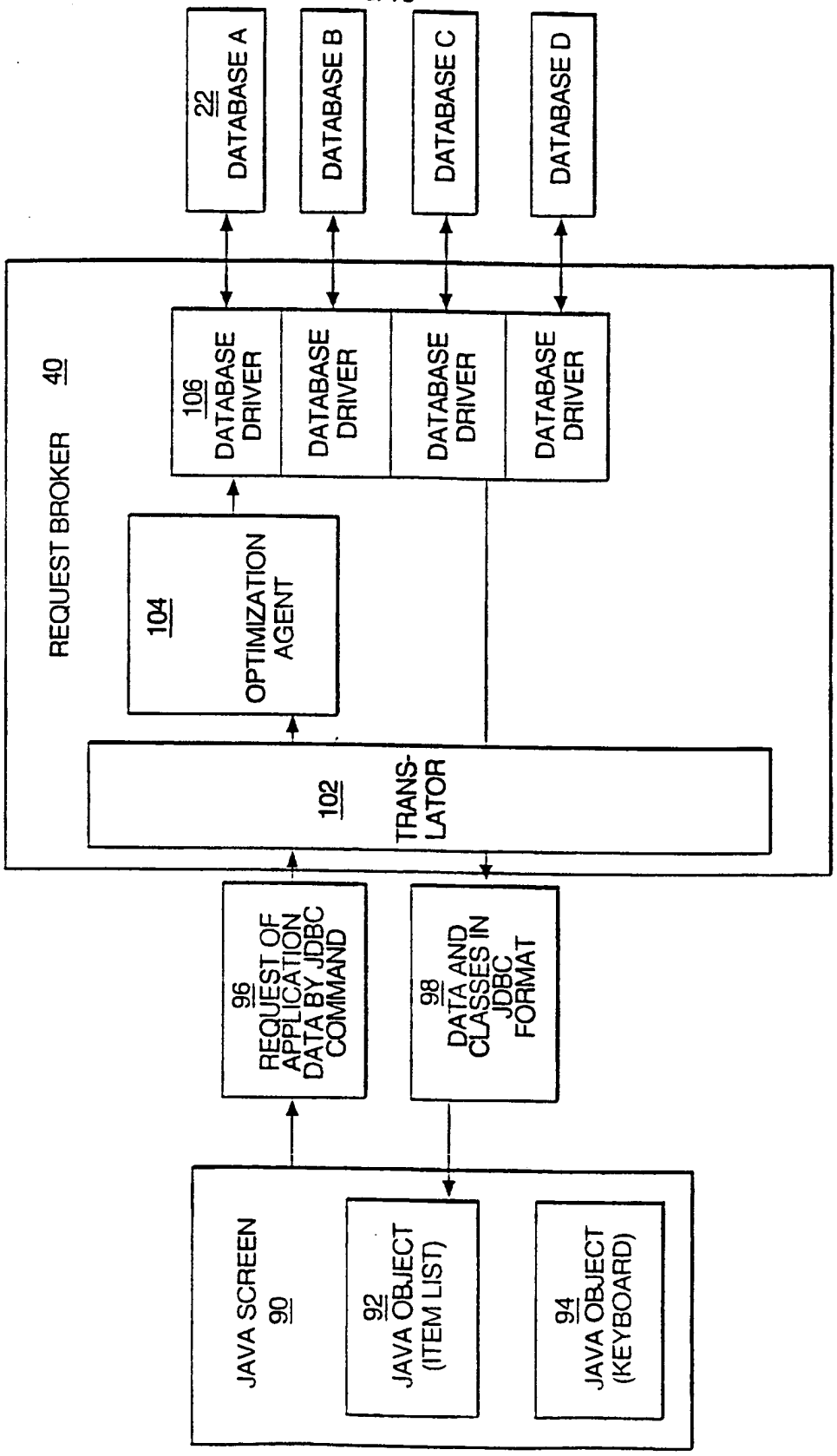

22 DATABASE A
DATABASE B
DATABASE C
DATABASE D
REQUEST BROKER 40
106 DATABASE DRIVER
DATABASE DRIVER
DATABASE DRIVER
DATABASE DRIVER
104 OPTIMIZATION AGENT
102 TRANS-LATOR
96 REQUEST OF APPLICATION DATA BY JDBC COMMAND
98 DATA AND CLASSES IN JDBC FORMAT
JAVA SCREEN 90
92 JAVA OBJECT (ITEM LIST)
94 JAVA OBJECT (KEYBOARD)

FIG. 5C

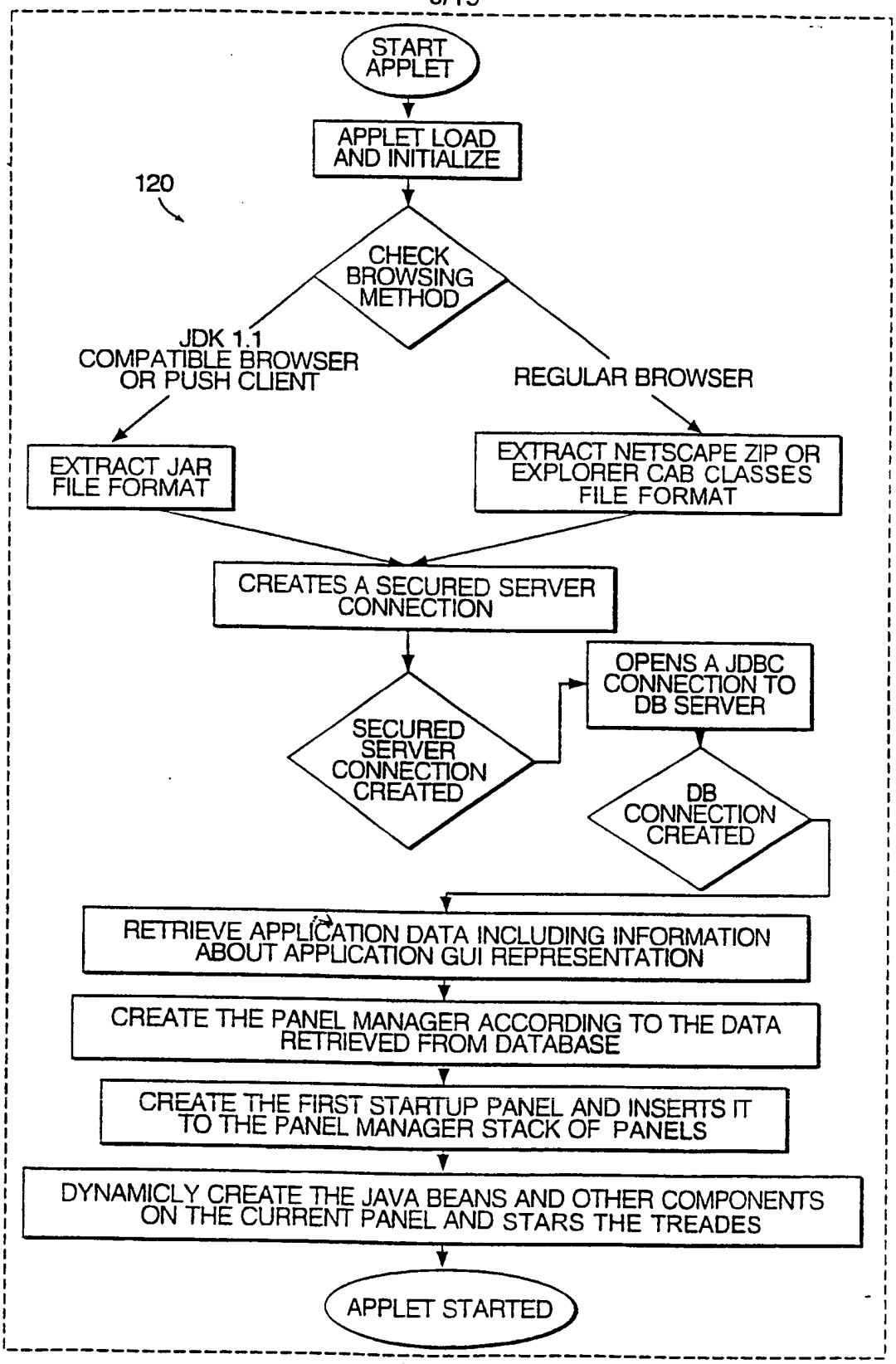
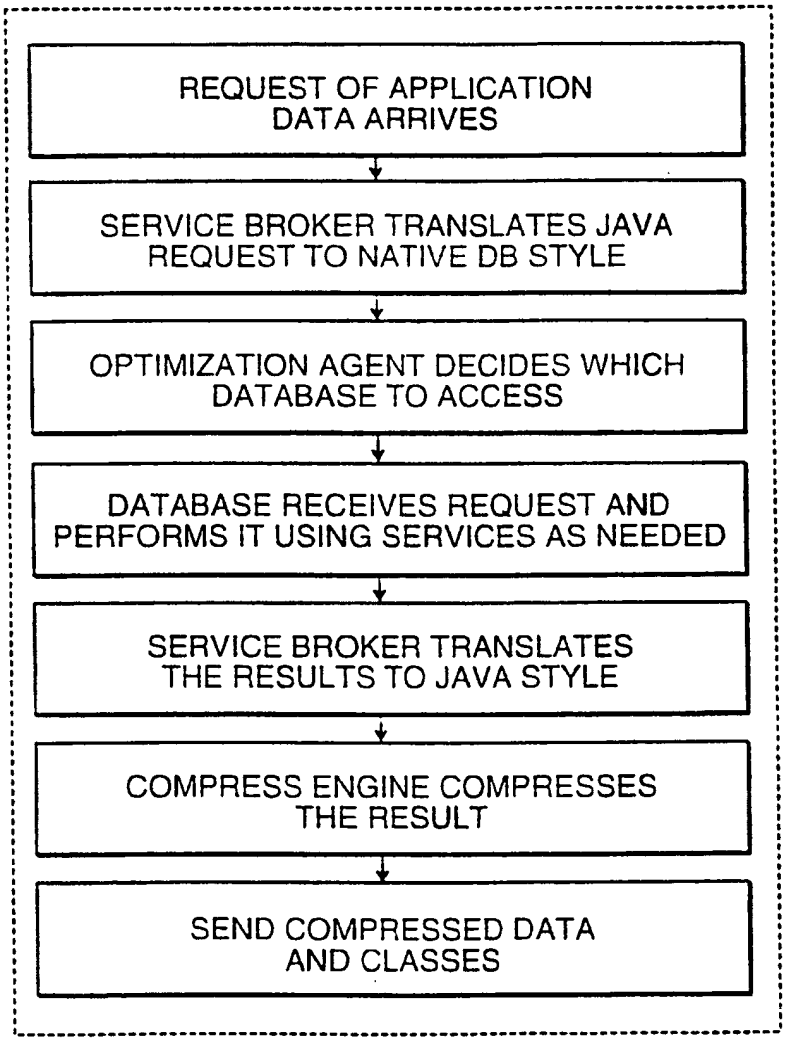

START APPLET
APPLET LOAD AND INITIALIZE
120
CHECK BROWSING METHOD
JDK 1.1 COMPATIBLE BROWSER OR PUSH CLIENT
REGULAR BROWSER
EXTRACT JAR FILE FORMAT
EXTRACT NETSCAPE ZIP OR EXPLORER CAB CLASSES FILE FORMAT
CREATES A SECURED SERVER CONNECTION
SECURED SERVER CONNECTION CREATED
OPENS A JDBC CONNECTION TO DB SERVER
DB CONNECTION CREATED
RETRIEVE APPLICATION DATA INCLUDING INFORMATION ABOUT APPLICATION GUI REPRESENTATION
CREATE THE PANEL MANAGER ACCORDING TO THE DATA RETRIEVED FROM DATABASE
CREATE THE FIRST STARTUP PANEL AND INSERTS IT TO THE PANEL MANAGER STACK OF PANELS
DYNAMICLY CREATE THE JAVA BEANS AND OTHER COMPONENTS ON THE CURRENT PANEL AND STARS THE TREADES
APPLET STARTED

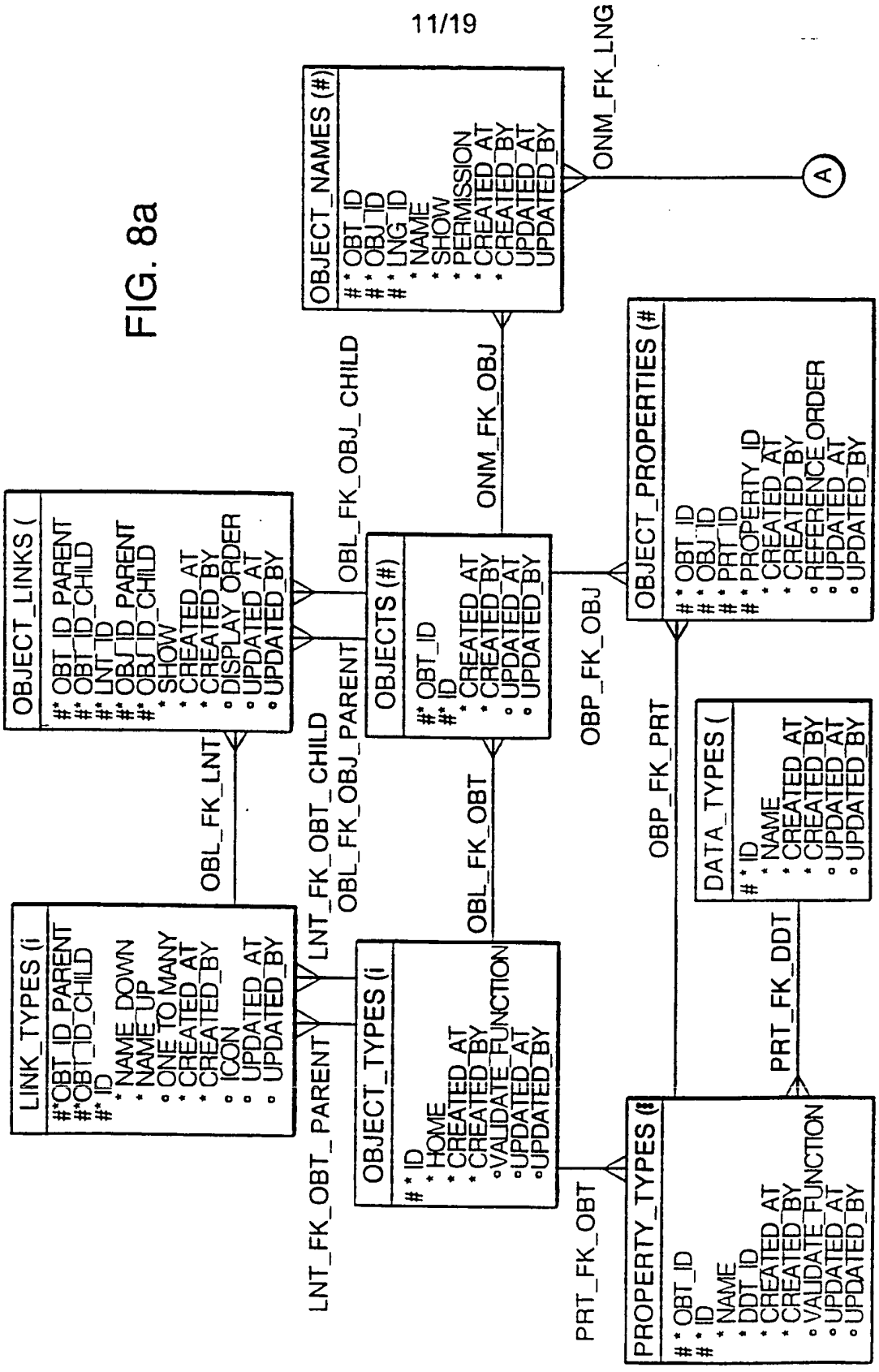
FIG. 6



FLOWCHART OF SERVER

FIG. 7

FIG. 8a


OBJECT_NAMES (#)
#* OBT_ID
#* OBJ_ID
#* LNG_ID
* NAME
* SHOW
* PERMISSION
* CREATED_AT
* CREATED_BY
UPDATED_AT
UPDATED_BY
ONM_FK_LNG
A
OBJECT_LINKS (
#* OBT_ID_PARENT
#* OBT_ID_CHILD
#* LNT_ID
#* OBJ_ID_PARENT
#* OBJ_ID_CHILD
* SHOW
* CREATED_AT
* CREATED_BY
DISPLAY_ORDER
UPDATED_AT
UPDATED_BY
OBL_FK_OBJ_CHILD
ONM_FK_OBJ
OBJECTS (#)
#* OBT_ID
#* ID
* CREATED_AT
* CREATED_BY
UPDATED_AT
UPDATED_BY
OBJECT_PROPERTIES (#
#* OBT_ID
#* OBJ_ID
#* PRT_ID
#* PROPERTY_ID
* CREATED_AT
* CREATED_BY
REFERENCE_ORDER
UPDATED_AT
UPDATED_BY
OBL_FK_LNT
LNT_FK_OBT_CHILD
OBL_FK_OBJ_PARENT
OBL_FK_OBT
OBP_FK_OBJ
OBP_FK_PRT
LINK_TYPES (i
#*OBT_ID_PARENT
#*OBT_ID_CHILD
#* ID
* NAME_DOWN
* NAME_UP
ONE_TO_MANY
* CREATED_AT
* CREATED_BY
ICON
UPDATED_AT
UPDATED_BY
OBJECT_TYPES (i
#* ID
* HOME
* CREATED_AT
* CREATED_BY
VALIDATE_FUNCTION
UPDATED_AT
UPDATED_BY
DATA_TYPES (
#* ID
* NAME
* CREATED_AT
* CREATED_BY
UPDATED_AT
UPDATED_BY
PRT_FK_DDT
LNT_FK_OBT_PARENT
PRT_FK_OBT
PROPERTY_TYPES (i
#* OBT_ID
#* ID
* NAME
* DDT_ID
* CREATED_AT
* CREATED_BY
VALIDATE_FUNCTION
UPDATED_AT
UPDATED_BY

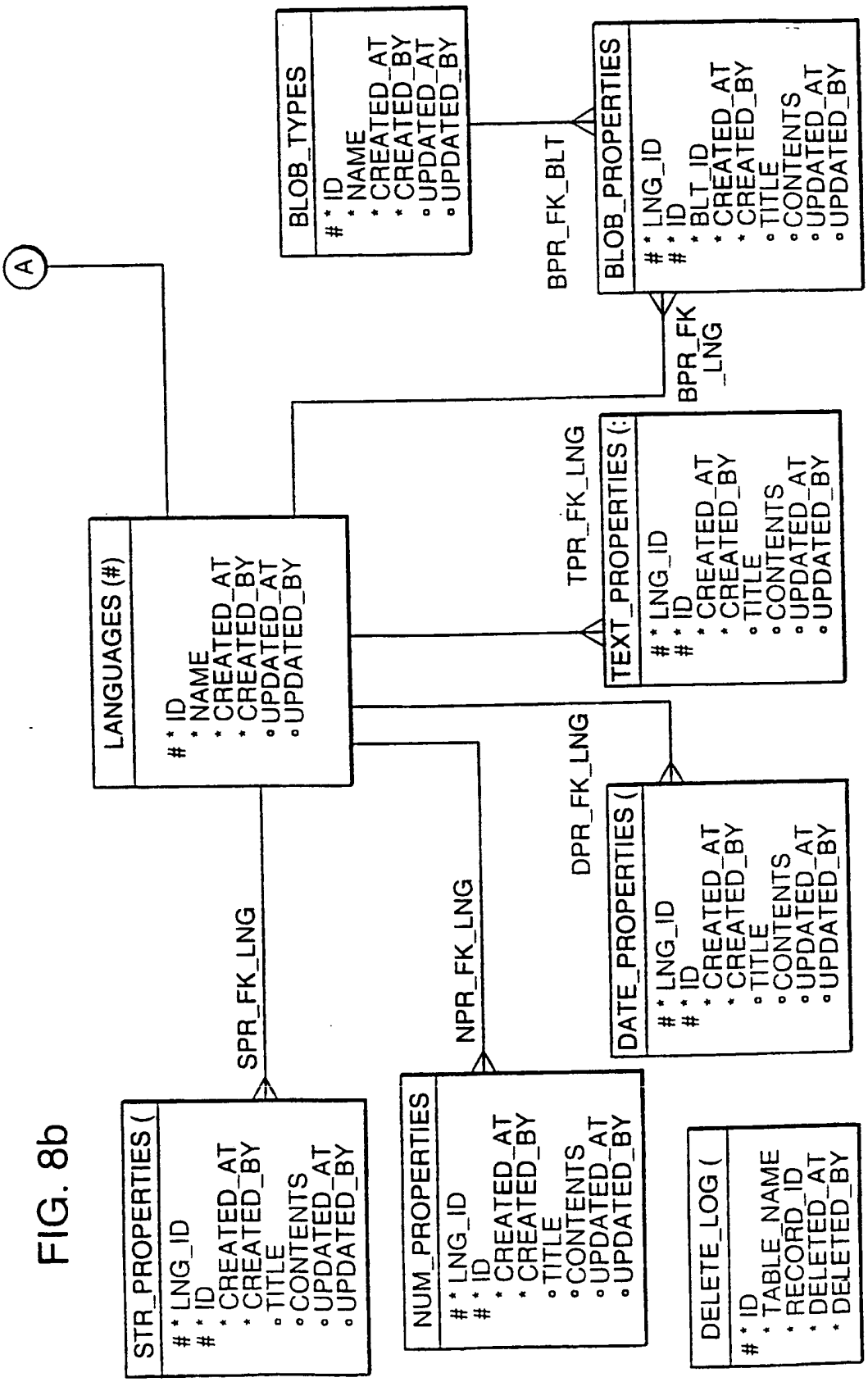

BLOB_TYPES
* ID
* NAME
* CREATED_AT
* CREATED_BY
◦ UPDATED_AT
◦ UPDATED_BY
BPR_FK_BLT
BLOB_PROPERTIES
* LNG_ID
* ID
* BLT_ID
* CREATED_AT
* CREATED_BY
◦ TITLE
◦ CONTENTS
◦ UPDATED_AT
◦ UPDATED_BY
A
BPR_FK_LNG
TPR_FK_LNG
TEXT_PROPERTIES (:
* LNG_ID
* ID
* CREATED_AT
* CREATED_BY
◦ TITLE
◦ CONTENTS
◦ UPDATED_AT
◦ UPDATED_BY
LANGUAGES (#)
* ID
* NAME
* CREATED_AT
* CREATED_BY
◦ UPDATED_AT
◦ UPDATED_BY
DPR_FK_LNG
DATE_PROPERTIES (
* LNG_ID
* ID
* CREATED_AT
* CREATED_BY
◦ TITLE
◦ CONTENTS
◦ UPDATED_AT
◦ UPDATED_BY
SPR_FK_LNG
NPR_FK_LNG
STR_PROPERTIES (
* LNG_ID
* ID
* CREATED_AT
* CREATED_BY
◦ TITLE
◦ CONTENTS
◦ UPDATED_AT
◦ UPDATED_BY
NUM_PROPERTIES
* LNG_ID
* ID
* CREATED_AT
* CREATED_BY
◦ TITLE
◦ CONTENTS
◦ UPDATED_AT
◦ UPDATED_BY
DELETE_LOG (
* ID
* TABLE_NAME
* RECORD_ID
* DELETED_AT
* DELETED_BY

FIG. 8b

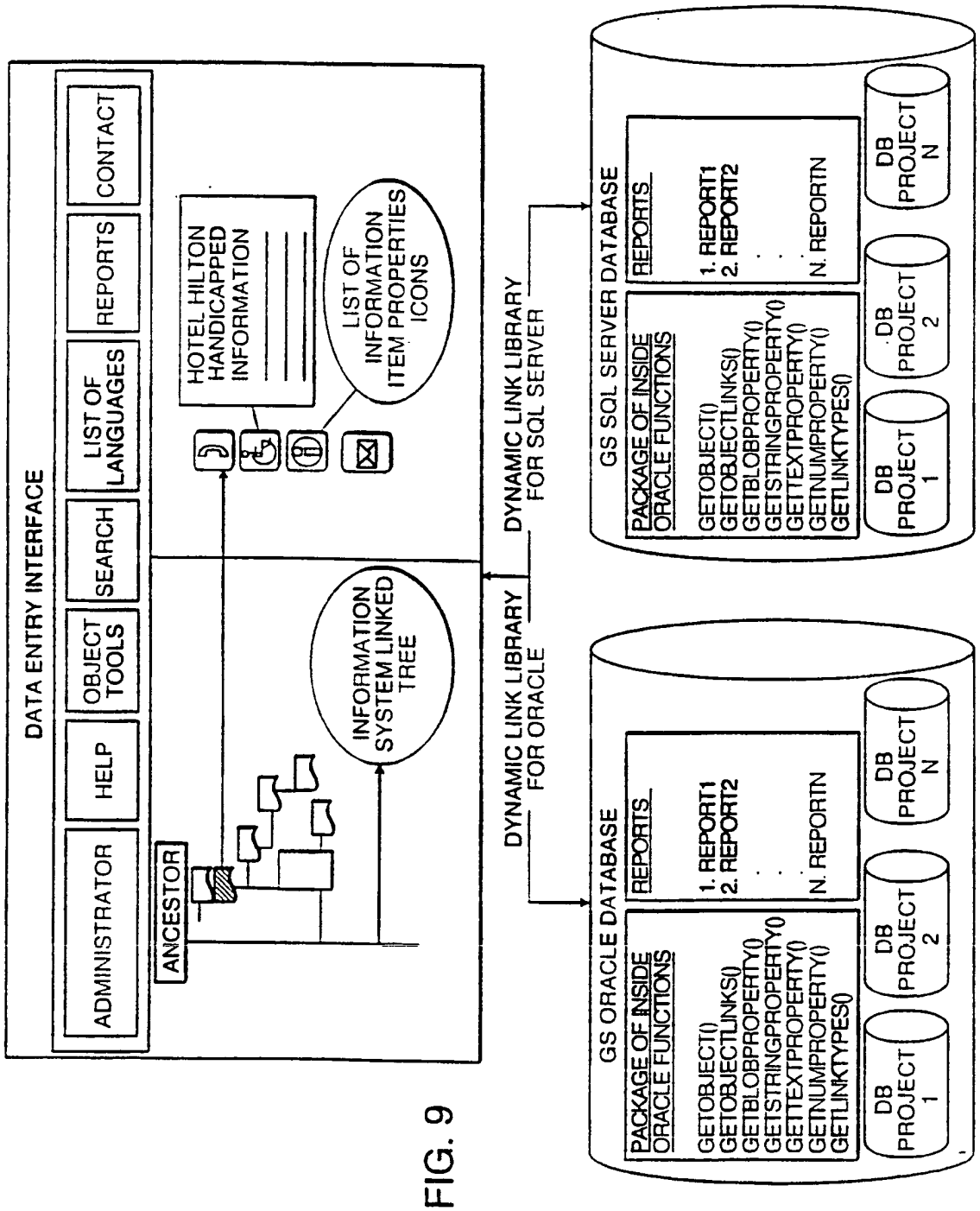

DATA ENTRY INTERFACE
ADMINISTRATOR
HELP
OBJECT TOOLS
SEARCH
LIST OF LANGUAGES
REPORTS
CONTACT
ANCESTOR
HOTEL HILTON HANDICAPPED INFORMATION
LIST OF INFORMATION ITEM PROPERTIES ICONS
INFORMATION SYSTEM LINKED TREE
DYNAMIC LINK LIBRARY FOR ORACLE
DYNAMIC LINK LIBRARY FOR SQL SERVER
GS ORACLE DATABASE
PACKAGE OF INSIDE ORACLE FUNCTIONS
GETOBJECT()
GETOBJECTLINKS()
GETBLOBPROPERTY()
GETSTRINGPROPERTY()
GETTEXTPROPERTY()
GETNUMPROPERTY()
GETLINKTYPES()
REPORTS
1. REPORT1
2. REPORT2
N. REPORTN
DB PROJECT 1
DB PROJECT 2
DB PROJECT N
GS SQL SERVER DATABASE
PACKAGE OF INSIDE ORACLE FUNCTIONS
GETOBJECT()
GETOBJECTLINKS()
GETBLOBPROPERTY()
GETSTRINGPROPERTY()
GETTEXTPROPERTY()
GETNUMPROPERTY()
GETLINKTYPES()
REPORTS
1. REPORT1
2. REPORT2
N. REPORTN
DB PROJECT 1
DB PROJECT 2
DB PROJECT N

FIG. 9

OBJECT_TYPES			PROPERTY_TYPES			
ID	NAME	...	ID	NAME	Ddt_ID (DATA TYPE)	...
1	CLASSIFICATION	...	1	ICON	BLOB	...
2	COMPANY	...	1	COMPANY=S PROFILE	TEXT	...
			2	LOGO	BLOB	...
			3	VIDEO	BLOB	...
			4	ESTABLISHMENT DATE	DATE	...
3	PRODUCTS	...	1	DESCRIPTION	TEXT	...
			2	PICTURE	BLOB	...

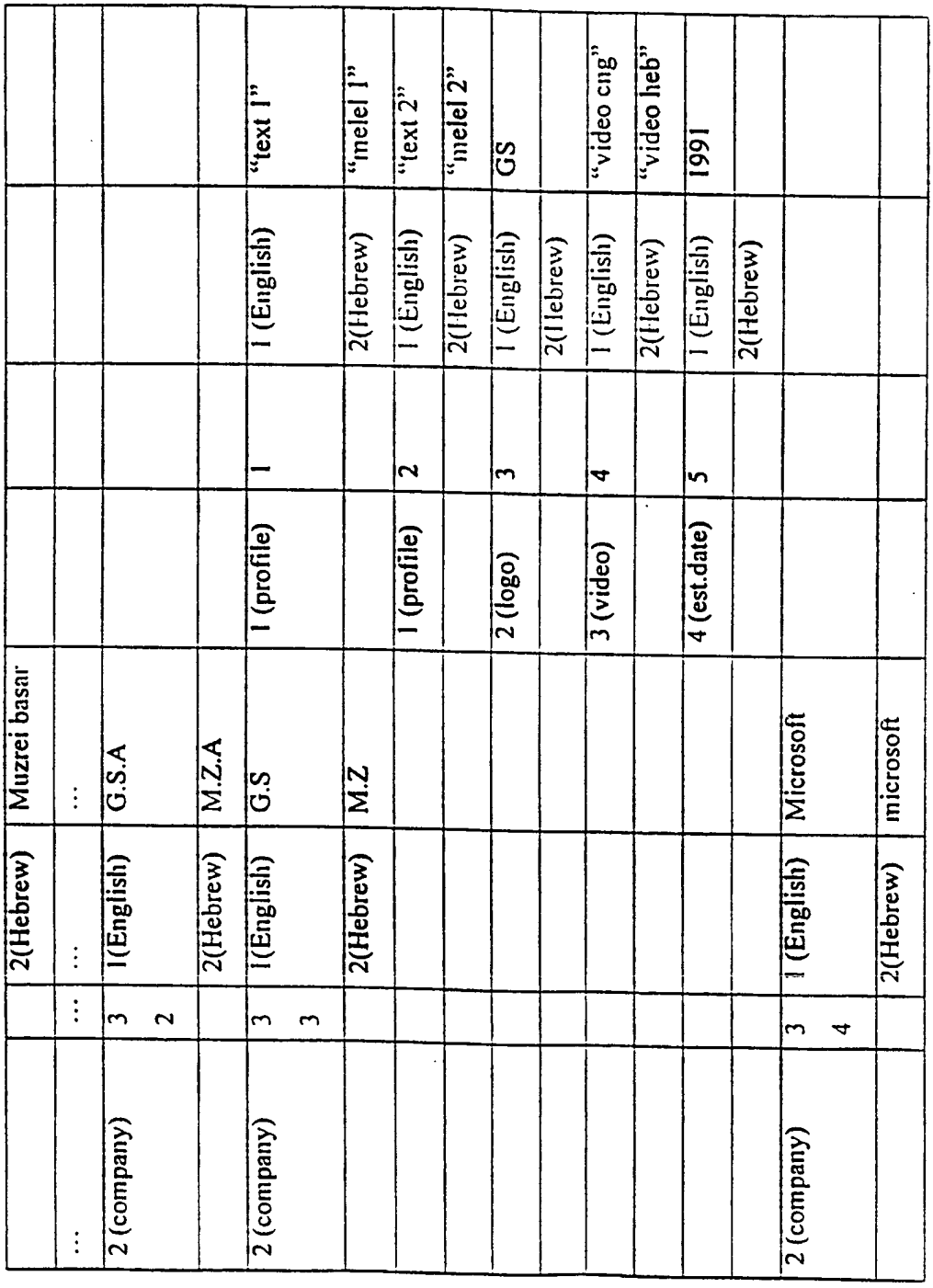
FIG. 10

LINK_TYPES					
OBT_ID_PARENT (PARENT OBJECT TYPE)	OBT_ID_CHILD (CHILD OBJECT TYPE)	ID	NAME_DOWN	NAME_UP	...
1 (CLASSIFICATION)	1 (CLASSIFICATION)	1	SUB-CLASSES	SUPER-CLASS	...
1 (CLASSIFICATION)	2 (COMPANY)	2	COMPANIES IN THIS CLASS	COMPANY=S CLASS	...
2 (COMPANY)	2 (COMPANY)	3	DAUGHTERS COMPANIES	MOTHER COMPANY	...
2 (COMPANY)	3 (PRODUCT)	4	COMPANY=S PRODUCTS	PRODUCT=S COMPANY	...

FIG. 11

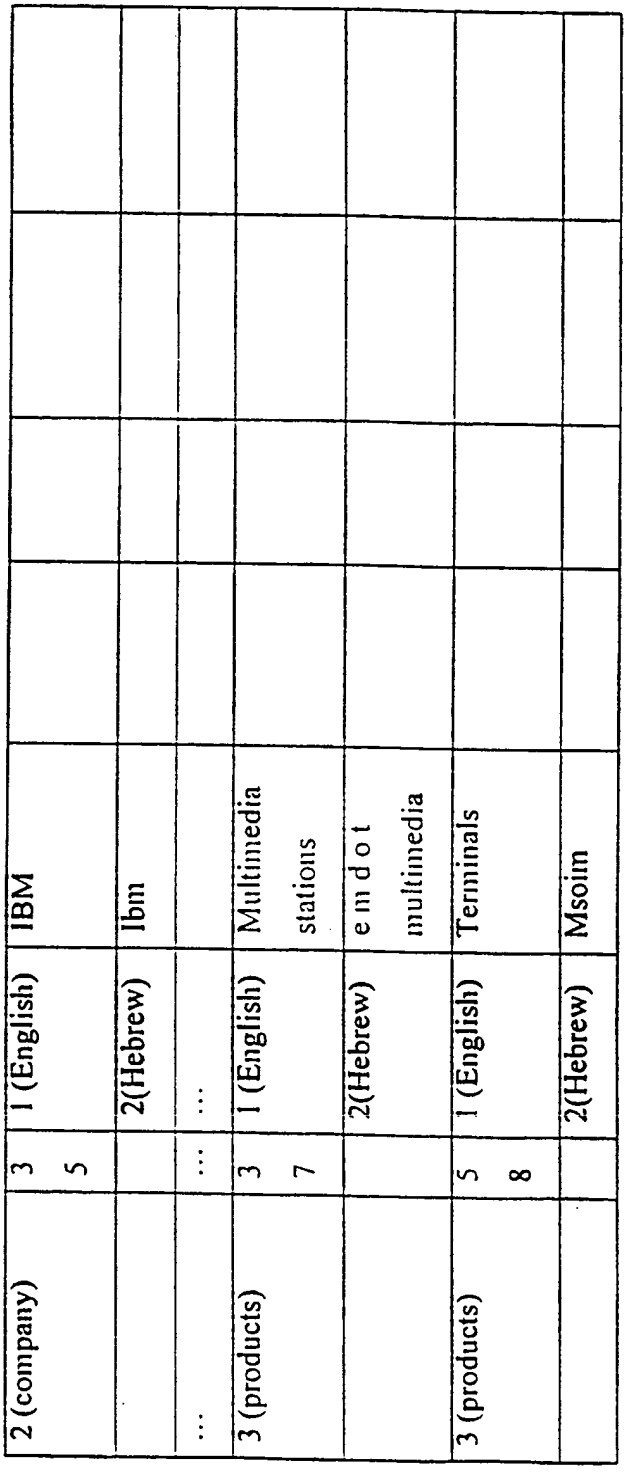
OBJECTS		OBJECT_NAMES		OBJECT_PROPORTIES		* PROPORTIES	
Obt_id (Object Type)	Id	Lng_id (language)	Name	Pft_id (proporty type)	Prop.id	Lng_id (language)	Contents
1(classification)	1	1(English)	Computers				
		2(Hebrew)	Machshevim				
1(classification)	2	1(English)	Software				
		2(Hebrew)	Tochna				
1(classification)	3	1(English)	Hardware				
		2(Hebrew)	Chomra				
...	..	...	...				
1(classification)	1 4	1(English)	Food				
		2(Hebrew)	Mazon				
1(classification)	1 5	1(English)	Dairy prod.				
		2(Hebrew)	Muzarei Halav				
1(classification)	1 6	1(English)	Meat prod.				

FIG. 12a



		2(Hebrew)	Muzrei basar						
...	...	...	...						
2 (company)	3	1(English)	G.S.A						
	2								
		2(Hebrew)	M.Z.A						
2 (company)	3	1(English)	G.S	1 (profile)	1	1 (English)	"text 1"		
	3								
		2(Hebrew)	M.Z			2(Hebrew)	"melel 1"		
				1 (profile)	2	1 (English)	"text 2"		
						2(Hebrew)	"melel 2"		
				2 (logo)	3	1 (English)	GS		
						2(Hebrew)			
				3 (video)	4	1 (English)	"video eng"		
						2(Hebrew)	"video heb"		
				4 (est.date)	5	1 (English)	1991		
						2(Hebrew)			
2 (company)	3	1 (English)	Microsoft						
	4								
		2(Hebrew)	microsoft						

FIG. 12b



2 (company)	3 5	1 (English)	IBM					
		2(Hebrew)	Ibm					
...	...	...						
3 (products)	3 7	1 (English)	Multimedia stations					
		2(Hebrew)	e m d o t multimedia					
3 (products)	5 8	1 (English)	Terminals					
		2(Hebrew)	Msoim					

FIG. 12c

ILLUSTRATION OF CLASSIFICATION TREE AND ASSOCIATED DATA ITEMS

THE CONTENTS OF OBJECT_LINKS TABLE IS ILLUSTRATED BELOW:

```
*
*
/COMPUTERS
*   *
*   /SOFTWARE                          -- TYPE 1 LINK
*   *   *
*   *   /GSA                           -- TYPE 2 LINK
*   *   *  *
*   *   *  /G                          -- TYPE 3 LINK
*   *   *  *
*   *   *     /MULTIMEDIA STATIONS     -- TYPE 4 LINKS
*   *   *     /...
*   *   *     /...
*   *   /G                             -- TYPE 2 LINK
*   *   *  *
*   *   *  /MULTIMEDIA STATIONS        -- TYPE 4 LINKS
*   *   *  /..
*   *   *  /...
*   *   *
*   *   /M                             - TYPE 2 LINK
*   *   *  *
*   *   *  /...
*   *   *  /...
*   *   *
*   *   I                              -- TYPE 2 LINK
*   *     *
*   *     /TERMINALS
*   *     /...
*   *     /...
*   *
*   /HARDWARE                          -- TYPE 1 LINK
*       *
*       I                          -- TYPE 2 LINK
*         *
*         /TERMINALS                   -- TYPE 4 LINK
*         /...
*         /...
*
/FOOD
  *
  /DAIRY PRODUCT
  *
  /MEAT PRODUCTS
```

FIG. 13

CORRECTED VERSION*

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 12/00

A1

(11) International Publication Number: WO 98/59460

(43) International Publication Date: 30 December 1998 (30.12.98)

(21) International Application Number: PCT/US98/12947

(22) International Filing Date: 22 June 1998 (22.06.98)

(30) Priority Data:
08/879,859 20 June 1997 (20.06.97) US

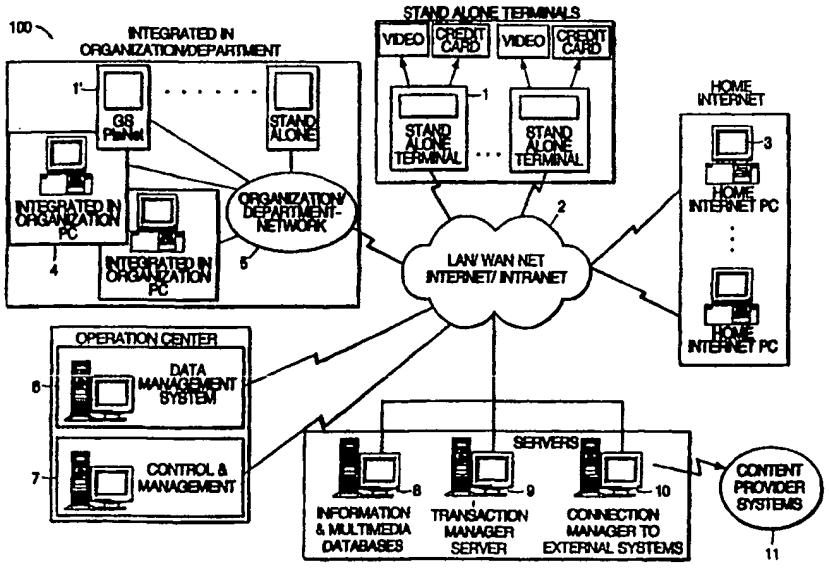
(71)(72) Applicant and Inventor: RAZ, Uri [US/US]; 36-02 Hillside Terrace, Fairlawn, NJ 07410 (US).

(74) Agent: SIBER, Victor; Fish & Richardson P.C., 45 Rockefeller Plaza, New York, NY 10111 (US).

(81) Designated States: CA, IL, JP, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: INFORMATION TRANSFER SYSTEM WITH DYNAMIC DISTRIBUTION OF DATA, CONTROL AND MANAGEMENT OF INFORMATION



(57) Abstract

A method and system (100) for dynamically distributing information across a network is provided. Client terminals (1, 1', and 3) and servers (8, 9, 10) are in data communication with a network (2), for example, the Internet. A database of characteristic data describing characteristics of the client terminals and the servers is generated by a data management system (6). The information residing on the network is dynamically distributed between the client terminals and the servers as a function of the characteristic data in the database, such that a portion of the data resides in the client terminals and in the servers.

*(Referred to in PCT Gazette No. 18/1999, Section II)

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US98/12947

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :HO4L 12/00
US CL :395/200.33

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 395/200.33

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A,E	US 5,802,306 A (HUNT) 1 Sept. 1998, abs., fig. 1.	1-38
A,E	US 5,799,318 A (CARDINAL et al.) 25 Aug. 1998, abs., fig. 1.	1-13
A,E	US 5,802,292 A (MOGUL) 1 Sept. 1998, abs., fig. 2.	1-38
A,E	US 5,802,296 A (MORSE et al.) 1 Sept. 1998, abs., figs. 1-2.	1-38

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 09 SEPTEMBER 1998	Date of mailing of the international search report 19 OCT 1998
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer LANCE LEONARD BARRY, ESQ. Telephone No. (703) 305-9600

Form PCT/ISA/210 (second sheet)(July 1992)*